

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Інститут прикладного системного аналізу
Кафедра математичних методів системного аналізу**

До захисту допущено
В. о. завідувача кафедри
_____ О.Л. Тимошук
«___» _____ 20__ р.

Дипломна робота

**на здобуття ступеня бакалавра
за освітньо-професійною програмою «Системний аналіз і управління»
спеціальності 124 «Системний аналіз»
на тему: «Моделі та оцінки можливих втрат на основі прогнозів волатильності»**

Виконав:

студент IV курсу, групи КА-61

Москаленко Володимир Володимирович _____

Керівник:

професор, д.т.н., професор кафедри ММСА

Бідюк Петро Іванович _____

Консультант з економічного розділу:

доцент, к.е.н., доцент кафедри ТТПЕ

Шевчук Олена Анатоліївна _____

Консультант з нормоконтролю:

доцент, к.т.н., доцент кафедри ММСА

Коваленко Анатолій Єпіфанович _____

Рецензент:

професор, д.т.н., професор кафедри ІБ ФТІ

Архипов Олександр Євгенович _____

Засвідчую, що у цій дипломній роботі
немає запозичень з праць інших авторів
без відповідних посилань.

Студент _____

Київ – 2020 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Інститут прикладного системного аналізу
Кафедра математичних методів системного аналізу

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 124 "Системний аналіз"

Освітньо-професійна програма «Системний аналіз і управління»

ЗАТВЕРДЖУЮ

В. о. завідувача кафедри

_____ О.Л. Тимощук

«___» _____ 20__ р.

ЗАВДАННЯ

на дипломну роботу студенту

Москаленку Володимирі Володимировичу

1. Тема роботи «Моделі та оцінки можливих втрат на основі прогнозів волатильності», керівник роботи Бідюк Петро Іванович, д.т.н., професор, затверджені наказом по університету від «25» травня 2020 р. №1143-с
2. Термін подання студентом роботи: «06» червня 2020 р.
3. Вихідні дані до роботи: статистичні дані стосовно розвитку вибраних фінансових гетероскедатичних процесів; типи математичних моделей для формального опису динаміки дисперсії; критеріальна база для аналізу адекватності моделей і якості оцінок прогнозів.
4. Зміст роботи: 1. Аналіз актуальності дослідження нестационарних процесів у фінансах та оцінюванні фінансових ризиків. 2. Типи математичних

моделей для опису динаміки волатильності фінансових процесів. 3. Обчислювальні експерименти стосовно побудови моделей та оцінювання прогнозів досліджуваних процесів. 4. Функціонально-вартісний аналіз розробки програмного забезпечення.

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо): – Мета, предмет та об'єкт дослідження; – Постановка задачі дослідження; – Типи математичних моделей для опису динаміки дисперсії; – Побудовані математичні моделі; – Результати обчислювальних експериментів.

6. Дата видачі завдання: «03» __09__ 2019 р.

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітка
1	Формулювання тематики (напряму) дослідження.	03.09.2019 – 30.09.2019	
2	Аналіз актуальності задач стосовно тематики дослідження	01.10.2019 – 30.10.2019	
3	Аналіз відомих результатів стосовно тематики дослідження	01.11.2019 – 30.11.2019	
4	Формулювання задач дослідження	01.12.2019 – 30.12.2019	
5	Уточнення теми дипломної роботи	25.02.2019	
6	Збір статичних даних, попередній аналіз даних	01.03.2020 – 30.03.2020	
7	Розробка програмного продукту для виконання обчислювальних експериментів	01.03.2020 – 30.04.2020	
8	Виконання обчислювальних експериментів, аналіз та оформлення результатів	01.05.2020 – 20.05.2020	
9	Оформлення пояснювальної записки у цілому	21.05.2020 – 31.05.2020	
10	Підготовка презентації для захисту	28.05.2020 – 01.06.2020	
11	Попередній захист дипломної роботи	01.06.2020 – 03.06.2020	
12	Захист дипломної роботи	15.06.2020 – 18.06.2020	

Студент

Володимир МОСКАЛЕНКО

Керівник

Петро БІДЮК

РЕФЕРАТ

Дипломна робота: 86 с., 14 табл., 23 рис., 2 дод., 20 джерел.

ВОЛАТИЛЬНІСТЬ, ГЕТЕРОСКЕДАСТИЧНІСТЬ, ІНТЕГРОВАНІЙ ЧАСОВИЙ РЯД, НЕЛІНІЙНИЙ ПРОЦЕС, НЕСТАЦІОНАРНИЙ ПРОЦЕС, ПРОГНОЗУВАННЯ, ФІНАНСОВИЙ РИЗИК.

Об'єкт дослідження: нелінійні нестационарні процеси в економіці та фінансах, фінансові ризики.

Предмет дослідження: моделі і методи аналізу нелінійних нестационарних процесів в економіці та фінансах, методи аналізу фінансових ризиків.

Мета дослідження: підвищення якості оцінок прогнозів нелінійних нестационарних процесів та їх волатильності, покращення оцінок ринкового ризику завдяки новим побудованим моделям.

Сфера застосування: нелінійні нестационарні процеси в галузі економіки та фінансів.

Проведено теоретичне дослідження фінансових ризиків, розглянутий перелік підходів до математичного моделювання, їх переваги та недоліки. Розглянуті сучасні моделі оцінювання нестационарних процесів.

Для оцінки процесів з трендом побудовано інтегровану модель авторегресії ковзного середнього. Також побудовано узагальнені лінійні моделі для прогнозування часових рядів зі змінною дисперсією.

Встановлено, що в результаті побудованих в рамках дослідження моделей, які оцінюють дисперсію, можна виконувати якісне прогнозування та динамічне оцінювання ризиків в режимі реального часу.

ABSTRACT

Thesis work: 86 pp., 14 tabl., 23 fig, 2 app., 20 sources.

VOLATILITY, HETEROSCEDASTICITY, INTEGRATED TIME SERIES, NONLINEAR PROCESS, NON-STATIONARY PROCESS, FORECASTING, FINANCIAL RISK.

Object of the study: nonlinear nonstationary processes in economy and finances, financial risks.

Subject of the study: models and methods for analysis of nonlinear nonstationary processes in economy and finances, financial risks, methods for analysis and estimation of financial risks.

The purpose of the research: improvement of forecasts quality for nonlinear nonstationary processes in economy and finances, financial risks and their volatility, improvement of estimates for market risk on the basis of the new constructed models.

Application field: analysis of nonlinear nonstationary processes in economy and finances.

The theoretical study of financial risks has been performed, and possible approaches to their mathematical modeling were considered; their advantages and drawbacks. Modern methods and technics for estimation of nonlinear nonstationary processes in economy and finances are presented.

To estimate the processes with trends an integrated autoregressive model with moving average was constructed. Also generalized linear models for forecasting time series with time varying variance have been built.

As a result of constructing of the models for variance estimation and forecasting it is possible to perform quality estimation and dynamic forecasting of financial risks in real time.

ЗМІСТ

	с.
СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ	9
ВСТУП	10
Розділ 1 ОСОБЛИВОСТІ РОЗВИТКУ МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ НЕСТАЦІОНАРНИХ ТА НЕЛІНІЙНИХ ПРОЦЕСІВ. АКТУАЛЬНІСТЬ ОЦІНЕННЯ ФІНАНСОВИХ ВТРАТ	
	12
1.1 Історичні досягнення, що передували сучасним дослідженням нелінійних та нестационарних процесів.....	12
1.2 Фінансові ризики	13
1.3 Актуальність дослідження.....	15
1.4 Підходи до математичного моделювання	15
1.4.1 Регресійний аналіз.....	16
1.4.2 Нейронні мережі.....	18
1.4.3 Узагальнені лінійні моделі	19
1.5 Комп'ютерні системи для побудови моделей фінансово-економічних процесів	21
1.5.1 Статистична програма EViews.....	21
1.5.2 Множина статистичних продуктів SAS	22
1.5.3 Статистичний пакет STATISTICA.....	23
1.6 Висновки до розділу.....	24
ПОСТАНОВКА ЗАДАЧІ	25
Розділ 2 МАТЕМАТИЧНІ МОДЕЛІ ДЛЯ ОЦІНЮВАННЯ ФІНАНСОВОГО РИЗИКУ	
	26
2.1 Огляд існуючих моделей	26
2.1.1 Прогнозування інтегрованих часових рядів	26
2.1.2 Моделі прогнозування часових рядів із змінною дисперсією	27
2.1.2.1 Модель ARCH.....	27
2.1.2.2 Модель GARCH.....	28
2.1.2.3 Модель EGARCH	29

2.1.2.4	Моделі TGARCH та GJR-GARCH	29
2.2	Статистичні тести для встановлення належності процесів до класу нелінійних нестационарних процесів	30
2.2.1	Тест Дікі-Фуллера	30
2.2.2	Тест KPSS (Kwiatkowski–Phillips–Schmidt–Shin)	31
2.2.3	Тест Уайта	32
2.2.4	Графічний метод	33
2.3	Методика побудови моделей гетероскедастичних процесів	34
2.4	Критерії адекватності моделі та якості оцінок прогнозів	35
2.4.1	Mean Squared Error	36
2.4.2	Sum Squared Error	36
2.4.3	Коефіцієнт детермінації R^2	36
2.4.4	t-статистика Стюдента	36
2.4.5	Критерій Маллоуза	37
2.4.6	Інформаційний критерій Акаїке	37
2.4.7	Коефіцієнт Тейла	38
2.5	Висновки до розділу	38
Розділ 3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ		40
3.1	Завдання та обґрунтування вибору способу реалізації	40
3.2	Опис даних для моделей	41
3.3	Побудова моделей та аналіз отриманих результатів	44
3.3.1	Реалізація моделі ARIMA	44
3.3.2	Реалізація моделей для роботи з гетероскедастичними процесами ..	50
3.4	Висновки до розділу	52
Розділ 4 ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ		53
4.1	Постановка задачі	53
4.2	Обґрунтування функцій програмного продукту	53
4.3	Обґрунтування системи параметрів програмного продукту	56
4.4	Кількісна оцінка параметрів	56

4.5	Аналіз експертного оцінювання параметрів.....	57
4.6	Аналіз рівня якості варіантів реалізації функцій	58
4.7	Аналіз рівня якості варіантів реалізації функцій	59
4.8	Економічний аналіз варіантів розробки програмного продукту	60
4.9	Висновки до розділу.....	63
	ВИСНОВКИ	64
	РЕКОМЕНДАЦІЇ	65
	ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	66
	ДОДАТОК А ПРОГРАМНИЙ КОД	68
	ДОДАТОК Б ІЛЮСТРАТИВНИЙ МАТЕРІАЛ	79

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ

АКФ	– Автокореляційна функція
АР	– Авторегресія
АРКС	– Авторегресії з ковзним середнім
ММП	– Метод максимальної правдоподібності
МНК	– Метод найменших квадратів
ПП	– Програмний продукт
УАРУГ	– Узагальнена авторегресійна умовно гетероскедастична модель
ЧАКФ	– Часткова автокореляційна функція
ANCOVA	– Analysis of covariance
ANOVA	– Analysis of variance
ARCH	– AutoRegressive Conditional Heteroscedasticity
ARIMA	– AutoRegressive Integrated Moving Average
ARMA	– AutoRegressive Moving Average
ECM	– Error Correction Model
EGARCH	– Exponential GARCH
GARCH	– Generalized AutoRegressive Conditional Heteroscedasticity
GJR-GARCH	– Glosten, Jagannathan and Runkle GARCH
MSE	– Mean Squared Error
SAS	– Statistical Analysis Systems
SSE	– Sum Squared Error
TGARCH	– Threshold GARCH
VaR	– Value at Risk

ВСТУП

Питання прогнозування супроводжує людство майже весь час. Прогнозування – це передбачення на основі наявних даних на пряму, характеру і особливостей розвитку та закінчення явищ і процесів у природі й суспільстві [1]. Завжди було важливим спрогнозувати стан хворої людини, знати майбутню поведінку погоди та, що є на сьогодні одним з ключових питань, отримати інформацію про майбутні економічні показники на основі поточних даних.

Фінансово-економічна ситуація, що властива чи не кожному підприємству та, загалом, світові, ніколи не буває повністю стабільною. На неї впливають такі фактори як політична ситуація, дії конкурентів, природні явища та безліч інших чинників, що можуть не мати прямого зв'язку, проте їх також треба враховувати. Таким чином, навіть врахувавши більшість факторів, завжди буде залишатись випадкова складова, яка буде впливати на результати прогнозування.

Протягом останніх десятиліть спостерігається зростаючий інтерес до побудови математичних моделей у фінансовій сфері. Це не дивно, адже за ці роки фінансовий ринок зазнав значних змін у своїй структурі, глобалізація економіки світу змінила характер поведінки цін та в цілому підхід до роботи з фінансовими процесами, що спровокувало інтенсивний розвиток пов'язаних з цим наукових напрямів, зокрема галузі часових рядів.

Провідні компанії залучають кращих науковців не лише задля максимально точного розрахунку прогнозних даних, а й щодо оцінки можливих втрат. Управління фінансовими ризиками відіграє все більшу роль у прийнятті рішень, саме тому побудова та дослідження математичних моделей для оцінювання ризиків в економіці та фінансах є надзвичайно актуальним питанням.

Об'єкт дослідження: нелінійні нестационарні процеси в економіці та фінансах, фінансові ризики.

Предмет дослідження: моделі і методи аналізу нелінійних нестационарних процесів в економіці та фінансах, методи аналізу фінансових ризиків.

Мета дослідження: підвищення якості оцінок прогнозів нелінійних нестационарних процесів та їх волатильності, покращення оцінок ринкового ризику завдяки новим побудованим моделям.

У першому розділі наведено огляд ключових теоретичних понять в царині фінансових ризиків, проаналізовано основні підходи до математичного моделювання та запропоновано програмні інструментарії для статистичних розрахунків.

У другому розділі розглянуто існуючі моделі для прогнозування розповсюджених типів часових рядів у економічній галузі та моделі оцінки фінансових ризиків. Запропоновано статистичні тести для наборів даних та критерії ефективності і якості моделей.

У третьому розділі обґрунтовано вибір рішень щодо створюваного програмного продукту для прогнозування та оцінки можливих втрат, описано застосування моделей, проведено обчислювальні експерименти та виконано аналіз отриманих результатів.

У четвертому розділі виконано функціонально-вартісний аналіз розробки програмного забезпечення.

Розділ 1 ОСОБЛИВОСТІ РОЗВИТКУ МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ НЕСТАЦІОНАРНИХ ТА НЕЛІНІЙНИХ ПРОЦЕСІВ. АКТУАЛЬНІСТЬ ОЦІНЕННЯ ФІНАНСОВИХ ВТРАТ

1.1 Історичні досягнення, що передували сучасним дослідженням нелінійних та нестационарних процесів

Роботи з аналізу часових рядів розпочалися зі стохастичних процесів. Перше фактичне застосування до даних процесів може бути віднесено до досліджень Юлета Дж.У. та Уолкера Г. у 1920–1930-х роках. В цей час значення ковзного середнього було введено для усунення періодичних коливань часового ряду, наприклад коливань через сезонність. Пізніше Герман Уолд представив моделі ARMA для стаціонарних рядів, але не зміг отримати функцію вірогідності для використання функції максимальної правдоподібності для оцінки параметрів. Вона з'явилася лише 1970 року. Тоді вийшла класична книга "Time Series Analysis: Forecasting and Control" Бокса Дж. та Дженкінса Г.М., що містить повну процедуру моделювання окремих рядів: специфікацію, оцінку, діагностику та прогнозування [2].

Сьогодні так звані моделі Бокса-Дженкінса використовуються, мабуть, найчастіше, і багато методів, що застосовуються для прогнозування та сезонного коригування, можна простежити за цими моделями.

В різних сферах життєдіяльності, наприклад, екологія, фінанси, економіка, часові ряди часто виявляють тенденцію до зростання, що говорить про їх нестационарність. У фінансовій сфері це дуже легко простежується в країнах з перехідною економікою, адже там наявна висока нестационарна динаміка розвитку.

Для представлення часових рядів, що описують динаміки різних процесів, використовують рівняння (регресійні, різницеві). У 1980-х роках було проведено чимало досліджень щодо характеристичного рівняння, де було

виявлено, що нестационарний часовий ряд може мати спільний одиничний корінь. Якщо нестационарний ряд можна зробити стаціонарними, взяттям різниць деякого порядку від вихідного часового ряду - це і є модель ARIMA (модель Бокса-Дженкінса).

Іншим напрямком розвитку часових рядів, що походить від моделей Бокса-Дженкінса, є нелінійні узагальнення, в основному моделі ARCH та GARCH. Ці моделі дозволяють параметризувати та спрогнозувати непостійну дисперсію, яка, в свою чергу, дуже часто наявна у фінансових часових рядів. За винахід цих моделей та запуск моделі виправлення помилок Грейнджеру У.Дж. та Енглу Р.Ф. у 2003 році дали Нобелівську меморіальну премію з економічних наук.

1.2 Фінансові ризики

Паралельно з розвитком математичних моделей для опису та прогнозування даних, вченими досліджувалися фінансові ризики. Формування самого поняття фінансового ризику можна віднести до моменту появи грошових відносин між людьми. Адже як в традиційній ситуації купівлі-продажу, так і в більш сучасній формі, наприклад, інвестиційна діяльність, наявна невід'ємна частина - ризик.

Поняття ризику є важливим у ряді наукових галузей, проте не існує єдиної загальноприйнятої думки щодо того, як його визначати та тлумачити.

Термін "ризик" використовується в багатьох напрямках і надаються різні визначення, залежно від галузі використання та контексту. Спільним для більшості розповсюджених формулювань є невизначеність та можливість небажаних результатів.

Як і поняття ризику, в літературі зустрічаються різні підходи до визначення поняття "фінансовий ризик". Наявні такі тлумачення фінансового ризику:

1. Фінансовий ризик описує ризик, коли компанія не має можливості виконати погашення боргу. Це може означати, що потенційні інвестори втратять гроші, що були інвестовані в компанію. Чим більше боргів у компанії, тим вищий потенційний фінансовий ризик [3].
2. Фінансовий ризик - це будь-яка загроза, яка гальмує фінансове зростання та прибутковість компанії [4].
3. Фінансові ризики – це спекулятивні ризики, для яких можливий як позитивний, так і обернений наслідок. Їх виділяє ймовірність отримання збитку в результаті заключення операцій, які змістовно, за своєю природою є ризикованими [5].
4. Фінансовий ризик - це ймовірність збитків внаслідок таких подій, як наприклад зміни ринкових цін [6].

Вимірювання фінансового ризику, тобто оцінка можливих втрат, включає як складову оцінку ринкових, кредитних, ліквідних та інших ризиків.

Ринковий ризик представляє ризики, які більшою мірою обумовлені змінними на ринку. Зазвичай сюди включають зміну курсу валют, процентні ставки, акції та ціни на товари.

Кредитний ризик - це ризик, що лежить в основі ризику дефолту контрагентів, починаючи від роздрібних клієнтів до торгових контрагентів. Тобто ризик недотримання фінансових зобов'язань через відстрочку платежу або неплатоспроможність. Управління ринковим та кредитним ризиками зазвичай здійснюється окремо в більшості фінансових установ.

Ризик ліквідності описує ситуацію, коли суб'єкт господарювання може зіткнутися з труднощами в реалізації активів або іншим способом залучення коштів для виконання фінансових зобов'язань.

На сьогодні наявна зростаюча тенденція розглядати комплексні ризики більш цілісно [7].

1.3 Актуальність дослідження

Стандартні моделі, що зазвичай використовуються сьогодні для опису даних, вже можуть давати досить точні прогнози. Проте реальність вказує на те, що справжні процеси в фінансовій сфері та економіці не набувають вигляду простих лінійних, чи стаціонарних моделей. Саме через цю складність ще не існує неперевершеної моделі, на яку можна повністю покладатися. Дійсні результати можуть як збігатися з прогнозом, так і мати несподівано різний фінальний розвиток. Працюючи з реальними грошима на фінансовому ринку, важливо знати наскільки ризикованим є та, чи інша операція, щоб приймати остаточне рішення.

Серед процесів, що зустрічаються в економіці, дуже розповсюдженими є саме гетероскедастичні процеси (процеси зі змінною в часі дисперсією). Хоча моделюванню нестаціонарних процесів вже була присвячена значна увага, зокрема, у дослідженнях Дрейпера Н. та Сміта Г. [8], Бідюка П.І. та Половцева О.В. [9], тема гетероскедастичних процесів ще не повністю розкрита і вимагає докладного розгляду.

Таким чином, оцінювання можливих втрат у нестаціонарних економічних та фінансових процесах є актуальною задачею сьогодення, яка потребує ґрунтовного наукового дослідження.

1.4 Підходи до математичного моделювання

За останні сто років в літературі була описана велика кількість математичних методів, що дозволяють спрогнозувати лінійні, нелінійні нестаціонарні та інші процеси. Всі методи прогнозування, як правило, розділяють на три широкі класи:

1. Прогнозування, за основу якої взято суб'єктивне судження, оцінка з урахуванням досвіту й інтуїції, поглиблені розуміння предметної

області та інші міркування, які мають пряме, або опосередковане відношення до досліджуваного процесу.

2. Методи, що ґрунтуються на застосуванні часового ряду однієї змінної. Тут можливі такі варіанти, як авторегресія (АР), авторегресії з ковзним середнім (АРКС), АРКС з урахуванням тренду та інші подібні моделі.
3. Векторні процеси, або ж часові ради декількох змінних, слугують як третій клас. Тут враховується залежність ендогенної змінної від деякого числа регресорів у правій частині.

Звичайно, що ці три класи не є взаємовиключними, тому цілком доцільне використання комбінацій двох, або навіть трьох приведених методів [10, с.8].

Розглянемо декілька популярних підходів до побудови математичних моделей.

1.4.1 Регресійний аналіз

Цей підхід включає в себе статистичний аналітичний метод, який враховує передбачувані відносини між залежною та однією або декількома незалежними змінними. Регресійний аналіз означає пошук такої функції f , яка описує вказаний зв'язок між змінними. Регресія може бути представлена як сума невинпадкової та випадкової складових:

$$\vec{y} = f(\vec{a}\vec{x}) + \varepsilon ,$$

де f – функція регресійної залежності;

ε – деяка випадкова величина;

\vec{x} та \vec{y} – відповідно значення незалежних та залежних змінних;

\vec{a} - набір параметрів.

Функція f може набувати різного вигляду. Розрізняють одновимірну, багатовимірну, параметричну, непараметричну регресію, яка може бути як лінійною, так і нелінійною.

Наведемо декілька простих прикладів регресійних моделей:

Лінійна регресія

$$y(k) = a_0 + a_1x(k) + \varepsilon(k)$$

Авторегресія

$$y(k) = a_0 + a_1x(k) + a_2x(k-1) + a_3x(k-2) + \varepsilon(k)$$

Нелінійна регресія:

Гіперболічна

$$y(k) = a_0 + \frac{a_1}{x(k)} + \varepsilon(k)$$

Поліноміальна

$$y(k) = a_0 + a_1x(k) + a_2x^2(k) + a_3x^3(k) + \varepsilon(k)$$

Коефіцієнти подібних рівнянь можна оцінювати звичайним МНК. Для моделей такого типу:

$$y(k) = a_0 + \frac{a_1a_2}{x(k)} + a_2e^{-bx(k)} + \varepsilon(k)$$

застосовують нелінійний МНК, метод максимальної правдоподібності ММП та інші методи [10, с. 16].

Звісно, звичайні моделі можна розширювати до виду множинної регресії, ускладнюючи їх. Зрозуміло, що в наведеному вище вигляді моделі рідко використовуються в реальному житті, адже вони досить прості та можуть виявитися неточними. Дуже часто застосовують їх комбінації, ускладнення та

інші модифікації. Однак не слід забувати, що моделям, які мають надлишкову складність, властиве поняття перенавчання. Тобто модель дає точні результати на тестових даних, проте виявиться непридатною для прогнозування.

Можна виділити такі переваги регресійного підходу:

1. Методи є відносно простими в плані реалізації.
2. Можна наочно побачити взаємозв'язки між залежними та незалежними змінними. Це, в свою чергу, надає змогу оцінювати значимість факторів та їх впливу.
3. Застосовують як для простих процесів, так і для процесів із складними взаємозв'язками.

1.4.2 Нейронні мережі

Останні дослідження показують, що нейронні мережі – це дуже потужний засіб для класифікації та прогнозування, отже зростає інтерес до використання нейронних мереж для моделювання та прогнозування фінансових часових рядів. Було продемонстровано, що нейронні мережі можуть наближати будь-яку неперервну функцію, в той час як класичні методи, наприклад ARIMA, передбачають, що між входами та виходами існує лінійна залежність.

Створення нейронних мереж кардинально відрізняється від традиційних моделей тим, що неможливо власноруч відслідкувати кожен параметр моделі, оскільки нейронна мережа проходить процес навчання на прикладах (з вчителем, без вчителя, або змішаний спосіб), де сама налаштовує зв'язки. Цей процес є як позитивним моментом, оскільки таким чином метод є дуже гнучким, так і негативним, бо навчання моделі досить складний процес, що потребує багато часу та ресурсів. Не слід забувати, що нейронним мережам притаманне перенавчання, яке створює додаткові перешкоди в побудові точного прогнозу.

На рисунку 1.1 показана стандартна схема побудови рекурентної нейронної мережі, в якій на вхід подаються змінні, що вважаються значущими та значення похибки.

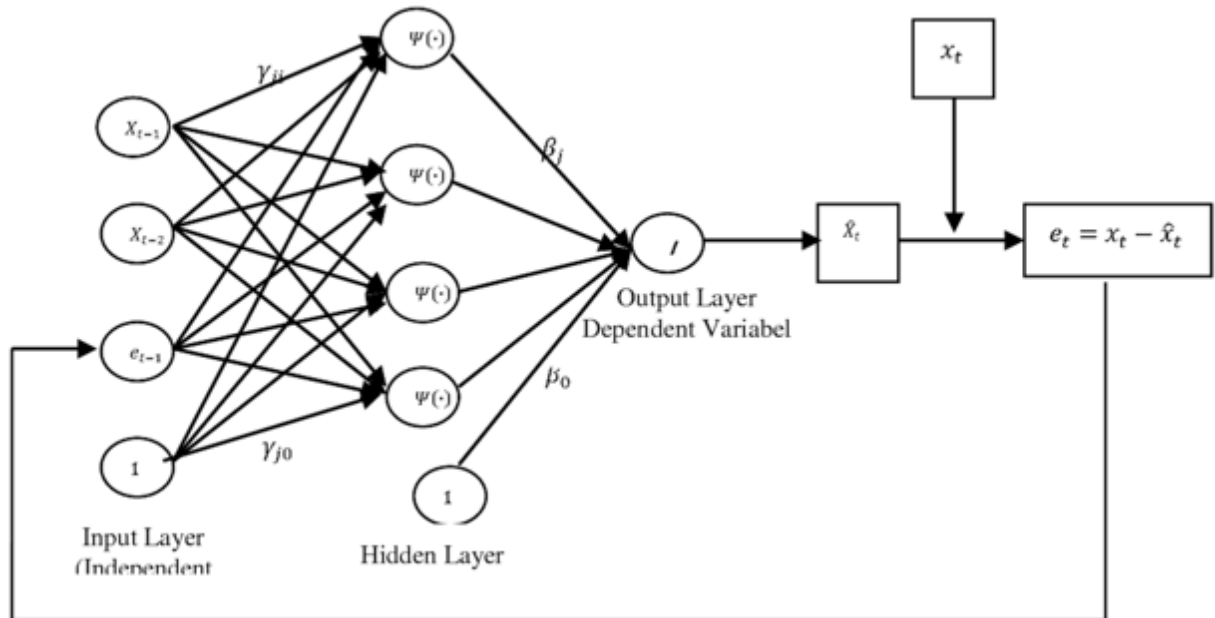


Рисунок 1.1 – Схема побудови рекурентної нейронної мережі [11]

В результаті навчання отримуються вагові коефіцієнти нейронів такі, що при отриманні нових даних мережа показувала коректний результат.

1.4.3 Узагальнені лінійні моделі

Узагальнені лінійні моделі були сформульовані Джоном Нелдером та Робертом Веддерберном як спосіб уніфікації різних статистичних моделей, включаючи лінійну регресію, логістичну регресію, регресію Пуассона та інших. Вони запропонували ітеративний зважений метод найменших квадратів для функції максимальної правдоподібності параметрів моделі.

Узагальнені лінійні моделі і стандартні лінійні моделі представляють собою залежну змінну Y

$$Y = \mu + \varepsilon ,$$

де μ – сума середнього значення;

ε – випадкова величина

Узагальнені лінійні моделі, на відміну від звичайних лінійних моделей, не передбачають нормальність, рівність дисперсій чи адитивність ефектів. Тут вводиться припущення про приналежність залежної змінної до експоненційного сімейства розподілів, допускається зміна дисперсії при зміні середнього. Очікувані значення відгуку представляють собою лінійну комбінацію предикторів, які пов'язані з залежною змінною через функцію зв'язку.

Загалом, існує три основні припущення для узагальнених лінійних моделей:

1. Припущення випадковості: всі компоненти відгуку Y незалежні, їх розподіл належить експоненціального сімейства.
2. Припущення систематичності: p предикторів об'єднуються в один лінійний предиктор

$$\eta = X\beta$$

3. Функція зв'язку $g(x)$: взаємозв'язок між припущеннями випадковості і систематичності виражається функцією зв'язку, яка є диференційованою та монотонною

$$\eta = g(E[Y])$$

У таблиці 1.1 наведені основні класи узагальнених лінійних моделей за Аланом Агресті.

Таблиця 1.1 – Основні класи узагальнених лінійних моделей [12]

Модель	Розподіл випадкової складової	Функція зв'язку	Систематична компонента
Лінійна регресія	Нормальний	Тотожна	Неперервна
Логістична регресія	Біноміальний	Логістична	Змішана
Лог-лінійна	Пуассона	Логарифмічна	Категоріальна
ANOVA	Нормальний	Тотожна	Категоріальна
ANCOVA	Нормальний	Тотожна	Змішана
Регресія Пуассона	Пуассона	Логарифмічна	Змішана

1.5 Комп'ютерні системи для побудови моделей фінансово-економічних процесів

Побудова будь-яких моделей сьогодні здійснюється за допомогою комп'ютерних програм. Комп'ютерні технології дозволяють нам швидко та точно виконати розрахунки. Проте зі збільшенням можливостей, ускладнюються й запити. Попри технічний розвиток і сучасні потужності, нам все ще потрібно витримувати баланс між швидкістю роботи та якістю результатів. Спеціальні програми можуть бути розроблені власноруч, або ж можна використовувати готові системи для статистичної роботи. Одними з найпоширеніших статистичних пакетів/програм є Eviews, SAS та STATISTICA.

1.5.1 Статистична програма EViews

EViews – це сучасний економетричний, статистичний та прогнозувальний інструментарій, який пропонує потужні аналітичні інструменти в рамках гнучкого, простого у користуванні інтерфейсу [13]. EViews представляє можливість швидкого та ефективного керування даними, проведення економетричного та статистичного аналізу, генерування прогнозів та побудову моделей, створення графіків та таблиць.

EViews – це потужний статистичний пакет, який широко використовується економістами, дослідниками, студентами та аналітиками для статистичного аналізу, моделювання і прогнозування досліджуваних процесів [14].

Використання EViews досить добре підходить для використання методу найменших квадратів, побудови узагальнених лінійних моделей, квантильної регресії, авторегресії з умовною гетероседастичністю, моделювання коінтегрованих процесів.

Перевагою EViews над іншими статистичними пакетами можна назвати простоту перенесення файлів з даними до програми. На відміну від інших, в EViews наявна можливість завантаження більшості популярних форматів. Також EViews пропонує такі функції, як інтерполяція параметрів даних та частотного фільтра, за допомогою яких можна генерувати пропущені дані. Завдяки цьому EViews є дуже зручним для аналізу часових рядів.

Попри корисність пакету EViews для роботи з часовими рядами, він не є корисним для проведення аналізу багатовимірних даних з довгостроковими наборами даних. Причиною тому є те, що пакет EViews обмежений матрицею ряду спостережень та змінних. Таким чином, отримання результатів для наборів багатовимірних даних може виявитися неможливим. З матричними операціями в оцінці ймовірності в EViews також можуть виникати проблеми. Для визначення функції ймовірностей для різних статистичних моделей треба спочатку записати вирази. Додатково вказувати детермінанти та квадратну форму для кожної моделі. Це завдання стає виснажливим під час роботи з двома або більш моделями.

1.5.2 Множина статистичних продуктів SAS

SAS (Statistical Analysis Systems) – це набір продуктів, створений Інститутом SAS, який виконує багатовимірні дослідження, сучасну аналітику, управління даними, бізнес-аналітику та багато інших завдань. SAS вже досить

довго використовує значна кількість великих організацій. Одним з зазначених продуктів є SAS Visual Statistics. Це система для інтерактивного аналізу даних, побудови прогнозів та описових моделей. Оскільки ця компанія ще з 1976 року на ринку, була створена потужна база для обробки даних та швидкодії цього процесу.

Можна виділити такі основні функціональні можливості SAS Visual Statistics: потужна техніка побудови моделей прогнозування, лінійної та логістичної регресії, узагальнених лінійних моделей, методи описового моделювання, кластеризація, дерева класифікації, порівняння моделі та їх оцінка. Зі сторони користувача треба зазначити, що працювати з цим інструментом можна не лише на персональному комп'ютері, а й у хмарних середовищах, що може стати надзвичайно корисним під час роботи з великими об'ємами даних.

Як і в кожній програмі, SAS Visual Statistics також має свої недоліки. Найбільш помітним є ціна продукту. Для великих підприємств покупка ліцензії не викликає труднощів, проте для власного користування ціна відіграє ключову роль. Наступним недоліком є складність освоєння, оскільки система дуже громізка та потребує високої кваліфікації користувача.

1.5.3 Статистичний пакет STATISTICA

Пакет STATISTICA, як і вищенаведені продукти, дозволяє виконувати основні задачі аналізу часових рядів. Головною перевагою можна сміливо назвати графічне представлення даних. Користувачу наданий великий спектр можливостей налаштування графіків та, загалом, процесу створення автоматичного звіту. Заявлено можливість роботи програми з необмеженими за розміром файлами, при цьому зберігаючи високу швидкість та точний результат. В пакеті наявний детально розроблений модуль для роботи з нейронними мережами, що на сьогоднішній день можуть давати високоякісні результати.

Попри те, що STATISTICA добре працює з часовими рядами, вона не є спеціалізованою за напрямком економетрики, тому відсутня можливість проведення деяких важливих статистичних тестів. Як і SAS, даний продукт має дуже високу ціну для звичайного користувача.

1.6 Висновки до розділу

В розділі було виконано теоретичне дослідження фінансових ризиків, розглянутий перелік підходів до математичного моделювання, їх переваги та недоліки. Надана характеристика та порівняння комп'ютерним системам для побудови моделей фінансово-економічних процесів та роботи з ними. Виявлені слабкі та сильні сторони існуючих систем, на які треба звернути особливу увагу при розробці власного програмного продукту.

Встановлено актуальність розв'язання задач аналізу фінансових процесів та динамічного оцінювання можливих втрат за наявності фінансових ризиків різних типів. Вказано на велику потребу у сучасних комп'ютерних системах для менеджменту ризиків. Визначено напрям подальших робіт для виконання дослідження за вибраною темою.

ПОСТАНОВКА ЗАДАЧІ

1. Виконати аналіз актуальності задач оцінювання ризиків фінансових втрат.
2. Зробити огляд математичних моделей для формального опису волатильності фінансових процесів.
3. Розробити програмний продукт для оцінювання ризиків можливих фінансових втрат (ринковий ризик).
4. Виконати обчислювальні експерименти стосовно математичного моделювання і динамічного прогнозування можливих фінансових втрат.
5. Надати пропозиції для виконання подальших досліджень за темою дипломної роботи.

Розділ 2 МАТЕМАТИЧНІ МОДЕЛІ ДЛЯ ОЦІНЮВАННЯ ФІНАНСОВОГО РИЗИКУ

2.1 Огляд існуючих моделей

2.1.1 Прогнозування інтегрованих часових рядів

Під час дослідження часових рядів доцільно оцінювати їх стаціонарність, тобто зробити перевірку на наявність одиничних коренів, порядок інтегрованості часового ряду. Якщо порядок інтегрованості більше нуля, то ряд перетворюється взяттям різниці відповідного порядку, доволі часто використовують перший або другий порядок. Для створеного стаціонарного ряду будується модель ARMA. Якщо модель вибрана правильно, то її випадкові помилки повинні приймати значення білого шуму. Загальний вигляд моделі ARIMA(p,d,q):

$$\Delta^d Y_t = c + \sum_{i=1}^p \alpha_i \Delta^d Y_{t-i} + \sum_{i=1}^q b_i \varepsilon_{t-i} + \varepsilon_t,$$

де Δ^d – оператор різниці часового ряду порядку d;

p – вказує на кількість лагів залежної змінної;

d – стільки разів береться різниця, щоб ряд став стаціонарним;

q - кількість лагів для значень ковзного середнього.

Дана модель також може використовуватися на часових рядах, в яких є яскраво виражена сезонність. Наприклад, взяття різниці для значень, що відповідають часу $t - 24$ або $t - 168$ дозволяє врахувати добову (або тижневу) періодичність. Досить розповсюдженою модифікацією моделі ARIMA є модель, що враховує сезонність, так звана SARIMA (Seasonal ARIMA) [15].

2.1.2 Моделі прогнозування часових рядів із змінною дисперсією

Дані моделі дуже часто застосовуються в економетриці для аналізу фінансових часових рядів. Причиною тому є той факт, що оцінки, одержувані в рамках звичайних моделей, не є в достатній мірі ефективними, оскільки не враховують явище гетероскедастичності, тобто змінної у часі дисперсії. Статистичний аналіз з використанням моделей ARCH передбачає, що шукане значення ціни в деякій мірі залежить від попередніх змін цін, тому їх вплив теж необхідно враховувати для оцінки поточного рівня. До того ж ARCH моделі припускають, що ступінь впливу попередніх значень цін знижується із збільшенням відстані від поточного моменту часу.

2.1.2.1 Модель ARCH

Припущення економетричних моделей полягає в тому, що даним притаманна гомоскедастичність (рівномірна дисперсія). Однак за деяких обставин це не так, і дослідник має працювати з гетероскедастичністю. На ринках цінних паперів періоди низької волатильності часто супроводжуються періодами підвищеної волатильності. Отже, дисперсія помилки, що описує ці ринки, відрізнялася б залежно від дисперсії попередніх періодів. Проблема гетероскедастичності полягає в тому, що довірчі інтервали можуть виявитися занадто вузькими, що покаже більшу точність, ніж це гарантується економетричною моделлю.

Авторегресійна умовно гетероскедастична модель була запропонована Енглом у 1982 році. Моделі ARCH намагаються моделювати дисперсію цих термінів помилок і в процесі виправляють проблеми, що виникають внаслідок гетероскедастичності. Мета моделей ARCH - забезпечити міру волатильності, яка буде прийнятною для формування фінансових рішень.

Рівняння для умовної дисперсії даної моделі порядку p має такий вигляд:

$$h(k) = \beta_0 + \sum_{i=1}^p \beta_i \varepsilon^2(k-i) + \varepsilon_1(k), \quad (2.1)$$

де $h(k)$ – умовна дисперсія обраного процесу в момент k ;

$\varepsilon^2(k)$ – квадрат залишків;

$\varepsilon_1(k)$, – похибка моделі в момент k [10, с.30].

Щоб не допустити негативних значень дисперсії приймається, що всі коефіцієнти моделі невід'ємні, а константа - додатна.

2.1.2.2 Модель GARCH

На основі моделі ARCH була запропонована узагальнена авторегресійна умовно гетероскедастична модель (УАРУГ). GARCH вже має три параметри, які дозволяють впливати на умовну дисперсію. Ця характеристика дозволяє GARCH бути лаконічнішою, ніж модель ARCH. Модель краще підходить для часових рядів, коли дані виявляють кластерність волатильності.

GARCH(p,q):

$$h(k) = \beta_0 + \sum_{i=1}^p \beta_i \varepsilon^2(k-i) + \sum_{i=1}^q \alpha_i h(k-i) + \varepsilon_1(k), \quad (2.2)$$

де q - кількість залишкових помилок у квадратах, які потрібно включити до моделі ARCH;

умовні позначення збігаються із відповідними позначеннями моделі за формулою 2.1.

Проте, як ARCH так і GARCH не зможе зафіксувати "ефекти фінансового важеля", які спостерігаються у фінансових даних. Ефект важеля викликаний тим, що негативна віддача має більший вплив на майбутню волатильність, аніж позитивна. Для вирішення цієї проблеми було запропоновано кілька розширень GARCH.

2.1.2.3 Модель EGARCH

Експоненційна модель УАРУГ була запропонована Нельсоном в 1991 році. Відмінність даної моделі полягає в тому, що змінною виступає не умовна дисперсія, а її логарифм. Саме цей аспект дає можливість в цій моделі працювати з ефектом важелю. Ще одне суттєве вдосконалення процесу EGARCH полягає в тому, що вона не містить обмежень на позитивну визначеність, адже логарифм може приймати й негативне значення [16].

Рівняння для логарифму умовної дисперсії даної моделі має такий вигляд:

$$\log(h(k)) = \beta_0 + \sum_{i=1}^p \beta_i \frac{|\varepsilon(k-i)|}{h(k-i)} + \sum_{i=1}^p \gamma_i \frac{\varepsilon(k-i)}{h(k-i)} + \sum_{i=1}^q \alpha_i \log(h(k-i))$$

2.1.2.4 Моделі TGARCH та GJR-GARCH

Порогова (Threshold) модифікація моделі GARCH була запропонована у 1994 році Закоюном. Її особливістю є те, що тут враховуються умовні стандартні відхилення. Данна модель може реагувати на позитивні та негативні зовнішні впливи асиметрично, що не прораховується у вихідних моделях.

TGARCH:

$$h(k) = \beta_0 + \sum_{i=1}^p \beta_i (|\varepsilon(k-i)| - \gamma_i I_{t-1} \varepsilon(k-i)) + \sum_{i=1}^q \alpha_i h(k-i) + \varepsilon_1(k)$$

$$I_t = \begin{cases} 1, & \varepsilon_t < 0 \\ 0, & \varepsilon_t \geq 0 \end{cases}$$

Паралельно із Закоюном, Глостен, Джаганатан та Ранк розробили свій варіант порогової моделі GARCH. Вона відрізняється від TGARCH

використанням умовної дисперсії, замість відхилень. В даній функції індикатор I_t враховує можливий вплив фінансового важеля (левереджу).

GJR-GARCH:

$$h(k) = \beta_0 + \sum_{i=1}^p (\beta_i + \gamma_i I_{t-1}) \varepsilon^2(k-i) + \sum_{i=1}^q \alpha_i h(k-i) + \varepsilon_1(k)$$

$$I_t = \begin{cases} 1, & \varepsilon_t < 0 \\ 0, & \varepsilon_t \geq 0 \end{cases}$$

2.2 Статистичні тести для встановлення належності процесів до класу нелінійних нестационарних процесів

Перед обранням прогнозуючої математичної моделі є доцільним проведення статистичних тестів для вибірки. Тести надають інформацію про стаціонарність, лінійність процесу. Наведемо декілька класичних тестів для визначення стаціонарності, гетероскедастичності та лінійності процесу.

2.2.1 Тест Дікі-Фуллера

Цей тест базується на перевірці наявності одиничного кореня (ознака нестационарності ряду). Розрізняють три варіанти тесту для моделі авторегресії

$$y(k) = ay(k-1) + \varepsilon(k)$$

Вважаємо $\Delta y(k) = y(k) - y(k-1)$, $p = a - 1$, p_0 та p_1 – константи.

1) Відсутні константа та тренд в моделі

$$\Delta y(k) = py(k-1) + \varepsilon(k)$$

2) Відсутній тренд в моделі, проте наявна константа:

$$\Delta y(k) = p_0 + p y(k-1) + \varepsilon(k)$$

3) В моделі наявна константа та тренд:

$$\Delta y(k) = p_0 + p_1 k + p y(k-1) + \varepsilon(k)$$

Параметри з наведених вище рівнянь оцінюються математичними методами, наприклад МНК. Критерієм тесту виступає перевірка параметра $p=0$. Якщо рівність виконується, то для визначення стаціонарності використовують спеціальну таблицю критичних значень статистики Дікі-Фуллера. Статистика Дікі-Фуллера – це t -статистика, але вона не розподілена за законом Стюдента, тому порівнювати значення треба саме з спеціальною таблицею. Виходячи з порівняння результатів з таблицею, нульова гіпотеза про нестаціонарність або приймається, або ж відхиляється.

2.2.2 Тест KPSS (Kwiatkowski–Phillips–Schmidt–Shin)

Даний тест також слугує для перевірки стаціонарності спостережуваного ряду. Припускається, що модель має вигляд

$$y_t = \delta t + \zeta_t + \varepsilon_t ,$$

де $\zeta_t = \zeta_{t-1} + u_t$ – випадкові блукання, а u_t - незалежні однаково розподілені випадкові величини ($E_{u_t} = 0, D_{u_t} = \sigma^2, \forall t$) [17, с.120];

ε_t – стаціонарний процес.

Критерій KPSS в якості нульової перевіряє гіпотезу про приналежність ряду до стаціонарних відносно тренду рядів (TS-рядів). Отже гіпотеза має вигляд

$$H_0: \sigma^2_{u_t} = 0$$

тобто ζ_t — константа, а статистика KPSS розраховується так:

$$KPSS = \frac{T^{-2} \sum_{t=1}^T \hat{S}_t^2}{\hat{\lambda}^2}$$

для якої

$$\hat{S}_t = \sum_{j=1}^t \hat{u}_j$$

це кумулятивна функція нев'язки, де $\hat{\lambda}$ — консистентна оцінка довгострокової дисперсії залишків

$$\hat{\lambda} = \lim_{T \rightarrow \infty} T^{-1} E(S_T^2)$$

Проаналізувавши значення статистики, можна робити висновок про стаціонарність процесу (наявність одиничних коренів).

2.2.3 Тест Уайта

В даному тесті відбувається перевірка на наявність гетероскедастичності в моделях лінійної регресії. Допускається, що процес гетероскедастичності може бути функцією однієї або декількох незалежних змінних. Для проведення тесту виконується оцінка регресії методом найменших квадратів (МНК), звідки беруться дані про залишки регресії. По квадратах цих залишків будується допоміжна модель регресії, використовуючи МНК. Допоміжна

регресія містить в собі константу, всі регресори, їх квадрати та попарні добутки. Її можна записати в такому вигляді:

$$e_t^2 = a_0 + a^T x_t + x_t^T A x_t + u_t ,$$

де e_t – залишки вихідної регресії;

x_t – її фактори;

u_t – випадкова величина, похибка допоміжної регресії;

a_0, a^T, A – параметри допоміжної регресії.

Далі висувається нульова гіпотеза про гомоскедастичність (відсутність гетероскедастичності). За статистику обирається nR^2 , де R^2 – коефіцієнт детермінації.

У випадку відсутності гетероскедастичності статистика буде асимптотично наближатись до $\chi^2(N - 1)$ – розподіл хі-квадрат, де N – кількість параметрів допоміжної регресії.

Тест підходить до широкого класу моделей та дає хороші результати на великих вибірках. Хоча тест Уайта забезпечує гнучку функціональну форму, яка корисна для виявлення майже будь-якого варіанту гетероскедастичності, вона не підходить для визначення способів виправлення чи коригування моделі гетероскедастичності [18]. Варто зазначити, що в більш конкретних випадках, за наявності уточнюючої інформації, використання спеціалізованих тестів може виявитися більш доцільним.

2.2.4 Графічний метод

Графічний метод базується на суб'єктивному судженні дослідника. В методі аналізуються залишки, або ж квадрати залишків моделі на графіку. Графічний аналіз залишків, є зручним в разі парної регресії. В разі множинної регресії подібний аналіз необхідно проводити для кожної значущої змінної окремо [19].

2.3 Методика побудови моделей гетероскедастичних процесів

1. Перше, що треба зробити – зібрати необхідні дані і провести попередню роботу з ними. Виконати нормування, заповнити відсутні виміри тощо. В додаток до графічного аналізу варто провести відповідні статистичні тести на гетероскедастичність. За наявності тренду, його, за необхідності, треба позбутися. В окремих випадках тренд може бути формально описаний разом з іншими складовими процесу.
2. На другому кроці будується авторегресія (або авторегресія з ковзним середнім) використовуючи показники автокореляційної функції (АКФ) та часткової АКФ для поданих даних. Таким чином обчислюємо ряд квадратів залишків $\{\hat{e}^2(k)\}$ та рахуємо його вибірккову дисперсію за:

$$\hat{\sigma}_e^2 = \frac{1}{N-1} \sum_{k=1}^N \hat{e}^2(k) ,$$

де N – число залишків в моделі AR (або ARMA), або ж це довжина ряду даних.

3. Далі обчислюється вибірккова автокореляційна функція для $\{\hat{e}^2(k)\}$ та будується графік (для наочності):

$$p(s) = \frac{\sum_{k=s+1}^N [\hat{e}^2(k) - \hat{\sigma}_e^2][\hat{e}^2(k-s) - \hat{\sigma}_e^2]}{\sum_{k=1}^N [\hat{e}^2(k) - \hat{\sigma}_e^2]}$$

Перевіряється значення $p(s)$. Якщо воно відрізняється від нуля (статистично), то тут наявний процес умовної гетероскедастичності. Щоб остаточно це перевірити застосовується статистика Лjung-Бокса:

$$Q = N(N + 2) \sum_{i=1}^{N/4} p(i)(N - i)$$

Статистика Q вказує на корельованість значень $\{\hat{e}^2(k)\}$. Якщо вони не корелюють, то Q -статистика буде розподілена за χ^2 з $N/4$ ступенями свободи.

4. На наступному етапі вже будується модель ARCH/GARCH чи їх модифікації по значенням $\{\hat{e}^2(k)\}$. На прикладі моделі GARCH за формулою 2.2, якщо існує коефіцієнт β_i ($i = 1, 2 \dots q$), що є значимим, то процес гетероскедастичний.
5. Отримавши оцінені значення залишків на 4-ому етапі ($\hat{e}_{\text{оц}}^2(k)$) можна уточнити вихідну модель так:

$$Y_{\text{уточ}}(k) = Y(k) - \hat{e}_{\text{оц}}(k)$$

На отриманому ряді будують модель AR або ARMA.

2.4 Критерії адекватності моделі та якості оцінок прогнозів

Після побудови декількох моделей-кандидатів, які відповідають модельованому процесу, треба визначити кращу. Для цього існують спеціальні критерії адекватності, точності та якості прогнозування. Сам термін адекватність означає, що модель має встановити коректну взаємодію та зв'язки між змінними процесу, правильно враховувати наявні шуми, можливі збурення та керуючі дії [10, с.24]. Найбільш розповсюдженими критеріями та показниками, що будуть прийнятними для фінансових моделей є MSE (Mean Squared Error), SSE (Sum Squared Error), коефіцієнт детермінації R^2 , t -статистика Стюдента, критерій Маллоуза, інформаційний критерій Акаїке та коефіцієнт Тейла.

2.4.1 Mean Squared Error

$$MSE = \frac{1}{n} \sum_{j=1}^n (Y_j - \hat{Y}_j)^2$$

2.4.2 Sum Squared Error

$$SSE = \sum_{j=1}^n (Y_j - \hat{Y}_j)^2$$

2.4.3 Коефіцієнт детермінації R^2

Цей коефіцієнт виступає показником інформативності ряду, порівнюючи дисперсії частини вихідного ряду та її наближення. Модель вважається адекватною, якщо $R^2 \rightarrow 1$. Проте на практиці значення вищі за 0.75 вважаються прийнятними.

$$R^2 = \frac{\sum_{j=1}^n |Y_j - \hat{Y}_j|^2}{\sum_{j=1}^n |\bar{Y}_j - Y_j|^2}$$

2.4.4 t-статистика Стьюдента

Дана статистика вказує на значущість оцінок параметрів моделі. За заданого рівня значущості α та числа ступенів свободи значення статистики звіряється з табличними.

$$t = \frac{(\bar{Y} - m)\sqrt{n}}{S_Y},$$

де m - деяке значення, що приймається за значення математичного сподівання в нуль-гіпотезі ($H_0: E[Y] = m$);

S_Y^2 – незміщена оцінка дисперсії.

$$S_Y = \sqrt{\frac{\sum_{j=1}^n (Y_j - \hat{Y}_j)^2}{n-1}}$$

2.4.5 Критерій Маллоуза

Основна ідея - порівняти здатність прогнозування деякої підмножини моделі та повної моделі. Повна модель, як правило, дає кращі результати, але якщо наявна мультиколінеарність, то оцінка параметрів може бути хибною.

У випадку, коли вибирається підмножина від повної моделі, що має менший показник мультиколінеарності, така модель може стати кращою, доки не буде значного зсуву у прогнозованих значеннях відносно повної моделі.

Користь для C_p полягає в тому, що ви можете використовувати її для вибору розміру моделі - отримання хорошої моделі, яка містить якомога менше змінних, зберігаючи однакові можливості прогнозування [20].

Критерій рахує відношення SSE для $(p-1)$ змінної моделі до MSE для повної моделі, а потім штрафує за кількість змінних:

$$C_p = \frac{SSE_p}{MSE} + 2p - n$$

Обирається модель, на якій критерій показує менше значення.

2.4.6 Інформаційний критерій Акаїке

Для оцінки адекватності використовують також критерій Акаїке. Даний критерій направлений на зменшення розходження між щільністю розподілу по обраній моделі та вихідній.

$$AIC = \frac{2(k-L)}{n}$$

$$L = -\frac{2}{n} \left(1 + \ln(2\pi) + \ln \left(\frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \right) \right),$$

де k – число параметрів моделі.

2.4.7 Коефіцієнт Тейла

Для визначення адекватності моделі з точки зору придатності для розв'язання задачі прогнозування використовують коефіцієнт неспівпадіння Тейла:

$$K_T = \frac{\sqrt{\sum_{i=1}^n \frac{(Y_i - \hat{Y}_i)^2}{n}}}{\sqrt{\sum_{i=1}^n \frac{Y_i^2}{n}} + \sqrt{\sum_{i=1}^n \frac{\hat{Y}_i^2}{n}}}$$

Прогноз вважається точним, якщо значення $K_T \rightarrow 0$

2.5 Висновки до розділу

В розділі розглянуто сучасні моделі для формального опису та оцінювання нестационарних процесів, що є дуже розповсюдженими у сфері фінансів, зокрема інтегровані часові ряди (ARIMA) та процеси зі змінною дисперсією (ARCH, GARCH, TGARCH, GJR-GARCH, EGARCH). Представлені переваги та недоліки використання моделей, враховуючи їх складність побудови та точність результатів. Були проаналізовані статистичні тести для отримання підтвердження гетероскедастичності, стаціонарності та лінійності спостережуваного процесу. Наведена загальна методика побудови моделей гетероскедастичних процесів. Для перевірки якості побудованих

моделей запропоновано ряд критеріїв адекватності моделі та якості оцінок прогнозів.

Встановлено, що дисперсія – основний параметр фінансових процесів, який забезпечує можливість оцінювання та прогнозування ризиків можливих втрат.

Розділ 3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Завдання та обґрунтування вибору способу реалізації

В рамках дипломної роботи потрібно було розробити програмне забезпечення для побудови моделей оцінок ризиків можливих фінансових втрат та виконати обчислювальні експерименти стосовно математичного моделювання і прогнозування можливих фінансових втрат.

Для розробки програмного забезпечення було використано середовище розробки Jupyter Notebook та мова програмування Python.

В основу розробки програми покладено регресійний підхід та узагальнені лінійні моделі, а саме моделі ARIMA, ARCH, GARCH, EGARCH.

Вибір моделі ARIMA обумовлений тим, що набори даних у фінансовій та економічній галузях досить часто містять в собі тренд. Для перевірки наявності тренду в даних використовується статистичний тест Дікі-Фуллера.

Також дуже розповсюджені вибірки даних зі змінною у часі дисперсією. Для таких даних, що мають високий рівень волатильності, застосовується тест Уайта, щоб підтвердити наявність гетероскедастичності. У разі підтвердження її наявності, використовуємо моделі ARCH GARCH EGARCH . Якщо гіпотеза про гетероскедастичність не підтверджується, використання настільки складних моделей не є доцільним і можна звертатись до простіших моделей, таких як AP або APKC.

Ризик можливих фінансових втрат оцінюємо із застосуванням методики VaR.

Для оцінювання адекватності побудованих моделей використовуються такі критерії ефективності як MSE, R^2 , інформаційний критерій Акаїке та критерій Дарбіна-Уотсона.

Додатково для оцінки якості застосовано порівняння з результатами, отриманими за допомогою програмного інструментарію EViews з використанням таких же наборів даних.

3.2 Опис даних для моделей

Для побудови прогнозу моделі використовують дані за попередні періоди. Для пошуку даних було використано інтернет-ресурс [kaggle.com](https://www.kaggle.com/), де у вільному доступі можна отримати релевантні дані у будь-якій галузі, в тому числі фінансово-економічній.

Дані у вибірці повинні бути однозначно визначеними, достовірними та повними. Тому перед завантаженням даних до програми було проведено попередню роботу щодо приведення їх до відповідності вище зазначеним критеріям.

В якості даних, що мають тренд, було обрано такі набори:

- Brent Oil Prices (Daily historical Brent Oil Prices available on the U.S. Energy Information Admin);
- China YOUZAN LTD (a payment tech company in mainland China)

Для моделювання процесів зі змінною дисперсією використано:

- Stock prices Dow Jones companies and DJIA dynamics (Info about the oldest US stock market index - Dow Jones Industrial Average)

В таблиці 3.1 наведено загальний опис наборів. В таблицях 3.2, 3.3, 3.4 наведено їх детальний опис.

Таблиця 3.1 – Опис наборів даних

Назва набору даних	Період	Обсяг
Brent Oil Prices	2010-08-04 – 2011-05-20	200
China YOUZAN LTD	2018-05-23 – 2018-11-13	118
DJIA	2017-01-02 – 2018-12-29	260

Таблиця 3.2 – Опис набору даних Brent Oil Prices

Дата	Ціна
2010-08-04	83.76
2010-08-05	82.9
...	...
2011-05-19	113.2
2011-05-20	111.25

Таблиця 3.3 – Опис набору даних China YOUZAN LTD

Дата	Ціна на момент закриття торгів
2018-05-23	1.00
2018-05-24	1.11
...	...
2018-11-12	0.47
2018-11-13	0.49

Таблиця 3.4 – Опис набору даних DJIA

Дата	Відносна ціна на момент закриття торгів
2017-01-02	0
2017-01-03	-0.382
...	...
2018-12-28	0.255
2018-12-29	-0.476

Графічне представлення відповідних наборів проілюстровано на рисунках 3.1, 3.2, 3.3.



Рисунок 3.1 – Графічне представлення даних Brent Oil Prices

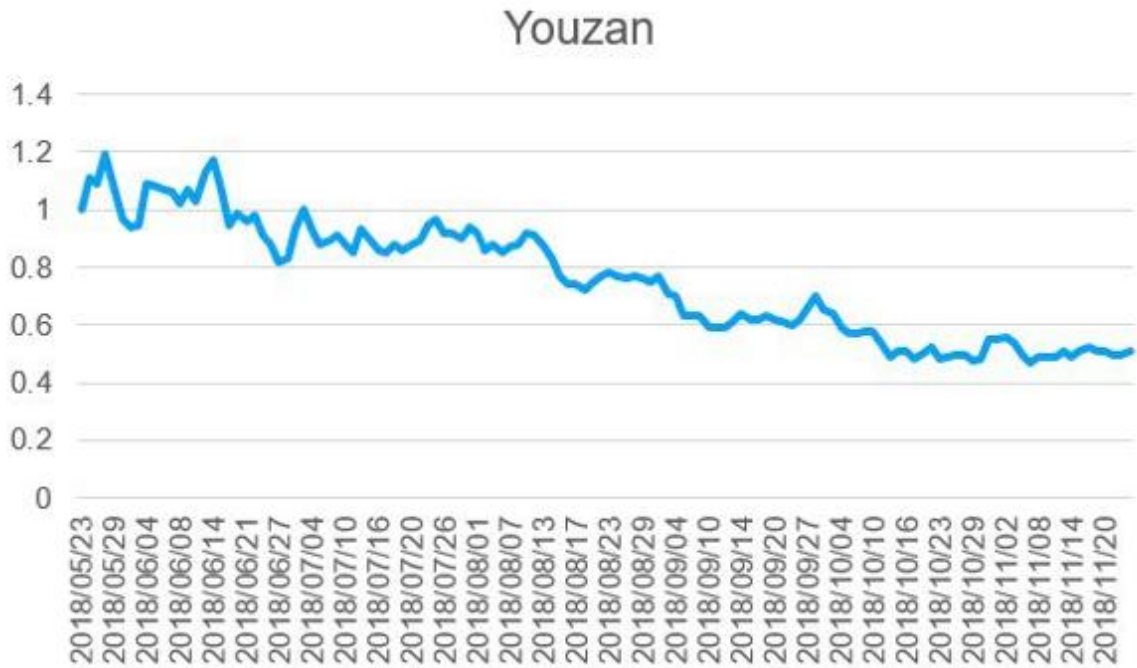


Рисунок 3.2 – Графічне представлення даних China YOUZAN LTD

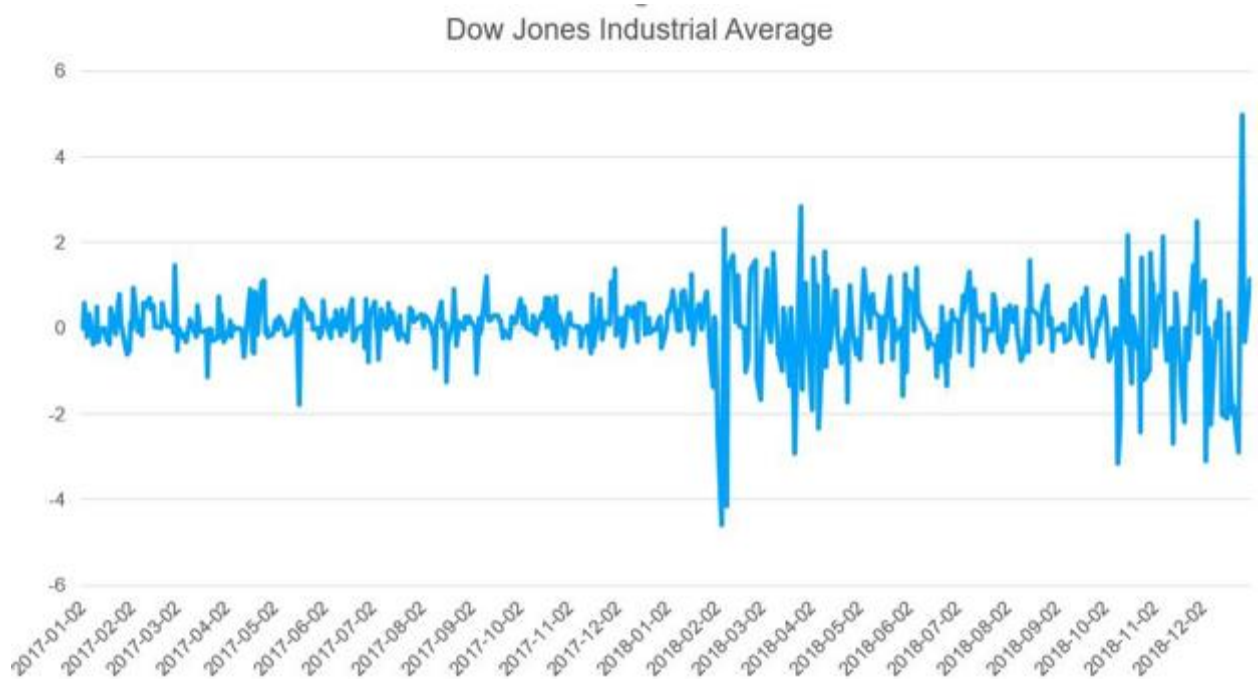


Рисунок 3.3 – Графічне представлення даних DJIA

3.3 Побудова моделей та аналіз отриманих результатів

3.3.1 Реалізація моделі ARIMA

Обираємо перший набір даних – Brent Oil Prices та проводимо попередній аналіз даних з використанням корелограмми (рисунок 3.4).

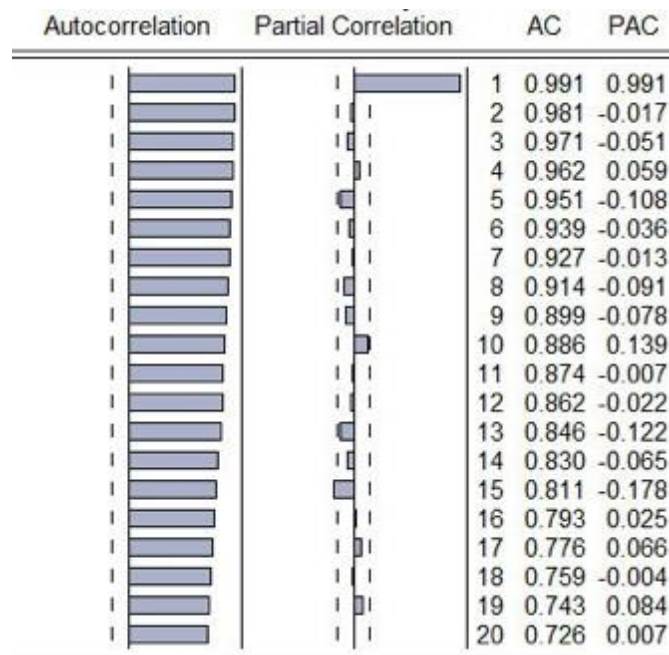


Рисунок 3.4 – Корелограма для обраних даних

Очевидно, що дані мають виражену взаємну залежність, тому виконано тестування даних на наявність тренду за допомогою Дікі-Фуллера. Результати тестування наведено на рисунку 3.5.

Null Hypothesis: BRENT_OIL has a unit root		
Exogenous: Constant, Linear Trend		
Lag Length: 0 (Automatic - based on AIC, maxlag=14)		
		t-Statistic
Augmented Dickey-Fuller test statistic		-2.912071
Test critical values:	1% level	-4.004836
	5% level	-3.432566
	10% level	-3.140059

Рисунок 3.5 – Результат тесту Дікі-Фуллера

Значення t-статистики Дікі-Фуллера знаходиться поза межами довірчого інтервалу, тому беремо різницю першого порядку і проводимо повторний тест. Результати повторного тестування наведено на рисунку 3.6.

Null Hypothesis: D(BRENT_OIL) has a unit root		
Exogenous: Constant, Linear Trend		
Lag Length: 0 (Automatic - based on AIC, maxlag=14)		
		t-Statistic
Augmented Dickey-Fuller test statistic		-14.79550
Test critical values:	1% level	-4.005076
	5% level	-3.432682
	10% level	-3.140127

Рисунок 3.6 – Результат повторного тесту Дікі-Фуллера

Після повторного тесту визначено, що t-статистика в межах довірчого інтервалу, тому зроблено висновок, що в даних наявний тренд першого порядку.

Отриманий ряд даних зображено на рисунку 3.7. На ньому бачимо, що тренд відсутній.

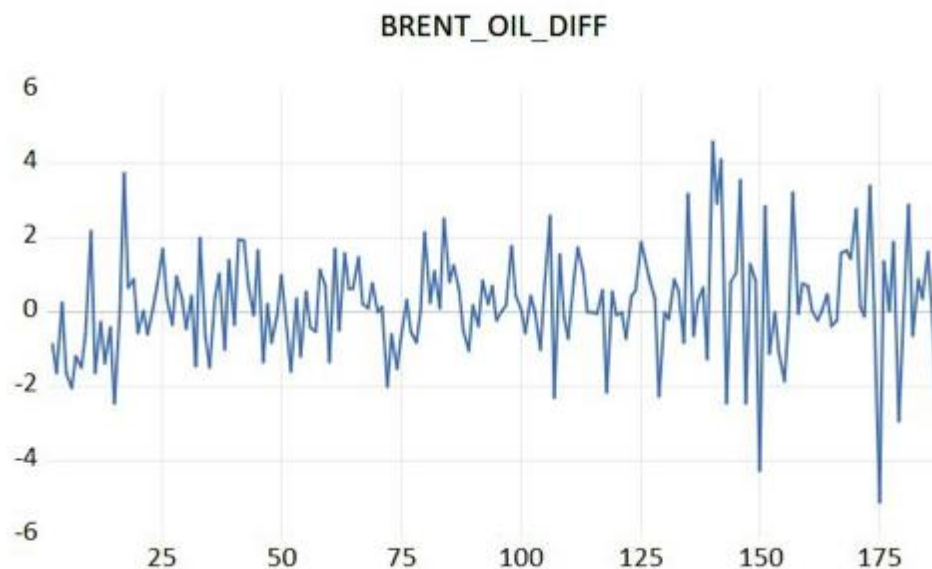


Рисунок 3.7 – Графічне представлення ряду перших різниць

Визначимо порядок моделі АРКС. Для цього використано порівняння значень ЧАКФ та отримано результат АРКС(1,4). Обчислені критерії

адекватності моделі порівнюємо із значеннями, отриманими з використанням EViews (таблиця 3.5).

Таблиця 3.5 – Критерії адекватності моделі

Назва критерію	Значення програми	Значення EViews
MSE	2.448	4.56
Інформаційний критерій Акаїке	-2.177	3.5
Коефіцієнт детермінації R^2	0.98	0.93
Критерій Дарбіна-Уотсона	2.03	2.08

Результати порівняння підтверджують достатній рівень достовірності роботи моделі. Таким чином, цю модель можна використовувати для прогнозування.

Виконуємо прогнозування ряду різниць першого порядку. (рисунок 3.8).

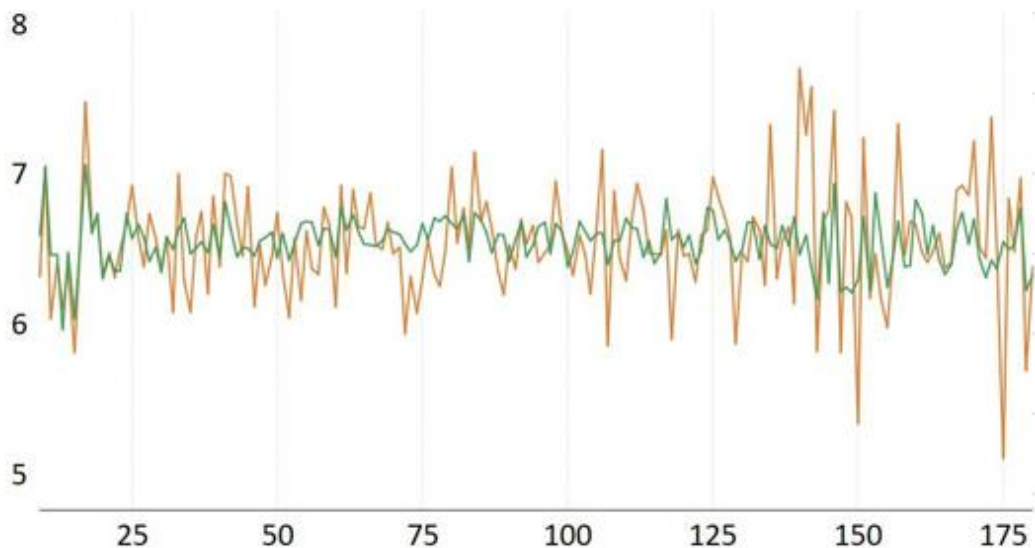


Рисунок 3.8 – Результат наближення моделі

З урахуванням отриманих результатів виконуємо прогнозування «ціни» та робимо оцінку можливих втрат. Розрахунок можливих втрат проводимо за формулою:

$$VaR_t^{1-\alpha} = k_{1-\alpha} \sigma_{t|t-1},$$

де $k_{1-\alpha}$ – квантиль заданого розподілу (зазвичай нормального);

$\sigma_{t|t-1}$ – волатильність прогнозованих значень;

$1 - \alpha$ – заданий рівень довіри (95%, 99% або інший).

Розраховані шляхом динамічного прогнозу дані та можливі ризики зображені на рисунку 3.10. На рисунку 3.10 також зображено фактичні дані за той же період часу.



Рисунок 3.10 – Прогнозні ціни(блакитний), можливі ризики(чорний) та фактичні дані(рожевий)

Проводимо аналогічні обчислювальні експерименти з набором даних Hong Kong Stock Prices (MTR). Результати наведені на рисунках 3.11-3.15 та в таблиці 3.6.

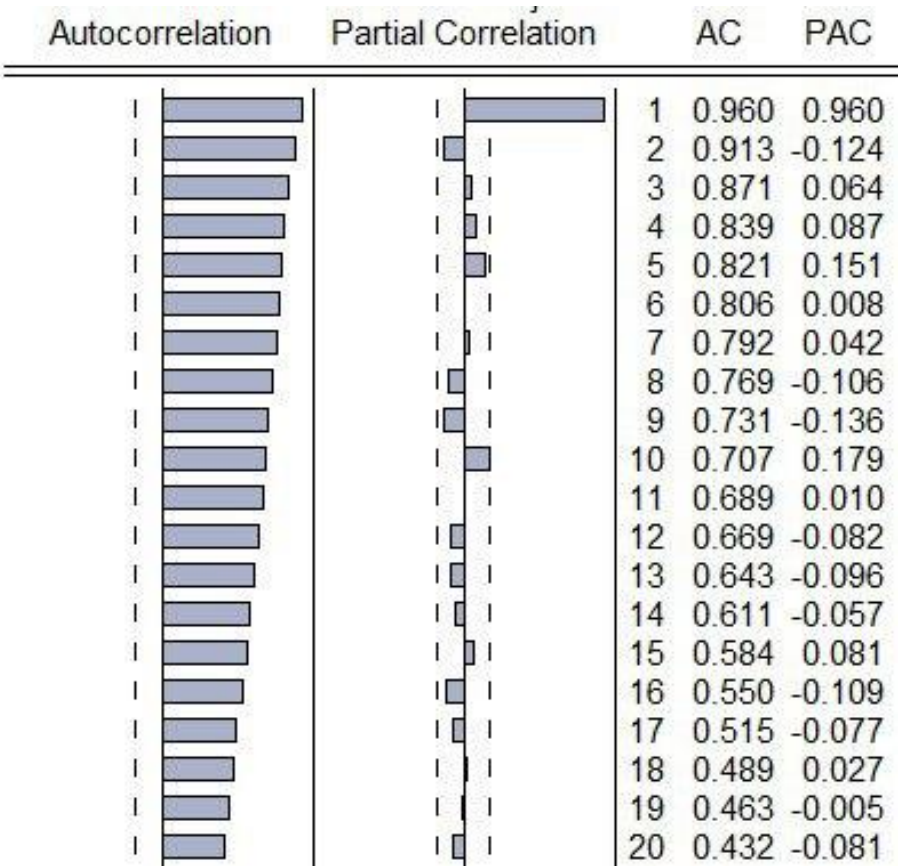


Рисунок 3.11 – Корелограма для обраних даних

t-Statistic		
Augmented Dickey-Fuller test statistic		
Test critical values:	1% level	-4.045236
	5% level	-3.451959
	10% level	-3.151440

Рисунок 3.12 – Результат тесту Дікі-Фуллера

t-Statistic		
Augmented Dickey-Fuller test statistic		
Test critical values:	1% level	-4.045236
	5% level	-3.451959
	10% level	-3.151440

Рисунок 3.13 – Результат повторного тесту Дікі-Фуллера (для ряду різниць першого порядку)

Таблиця 3.6 – Критерії адекватності моделі

Назва критерію	Значення програми	Значення EViews
MSE	0.00114	0.0129
Інформаційний критерій Акаїке	3.177	-3.4
Коефіцієнт детермінації R^2	0.93	0.87.
Критерій Дарбіна-Уотсона	1.91	1.88

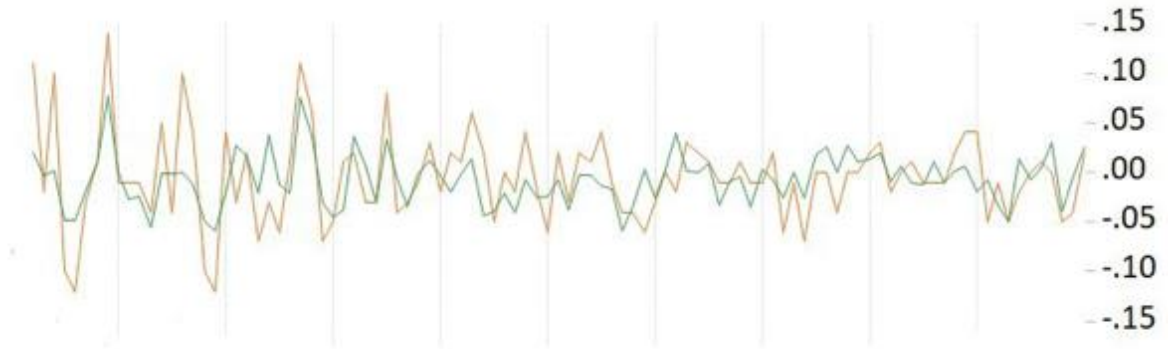


Рисунок 3.14 – Результат наближення моделі

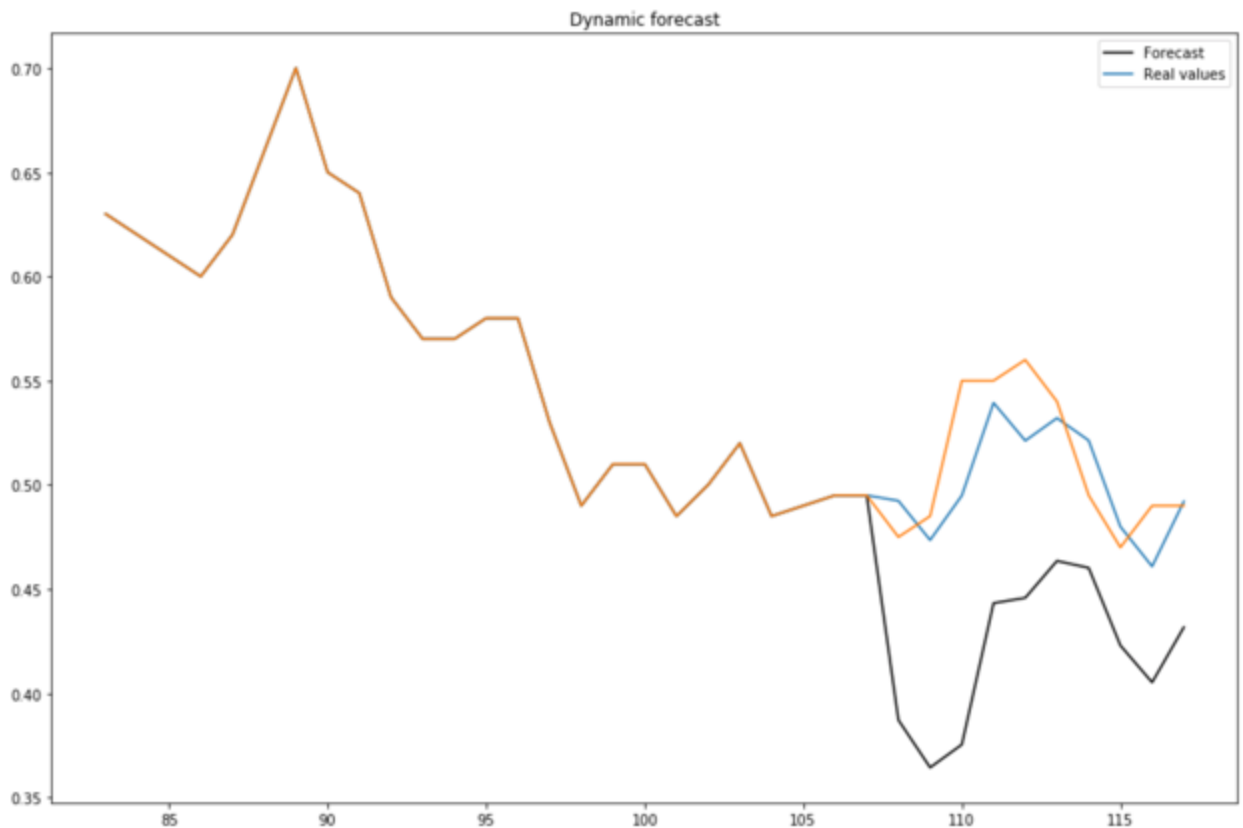


Рисунок 3.15 – Прогнозні ціни(блакитний), можливі ризики(чорний) та фактичні дані(рожевий)

3.3.2 Реалізація моделей для роботи з гетероскедастичними процесами

Обираємо набір даних – Stock prices Dow Jones companies and DJIA dynamics та проводимо аналіз на наявність гетероскедастичності з використанням тесту Уайта. Для проведення тесту виконується оцінка регресії методом найменших квадратів, звідки беруться дані про залишки регресії. Значення статистики тесту 0.002789, що менше за 0.004 (табличне значення розподілу χ^2 -квадрат), отже гіпотезу про гомоскедастичність відхиляємо. На рисунку 3.16 наведено графік вибіркової дисперсії даних за 2017 рік.

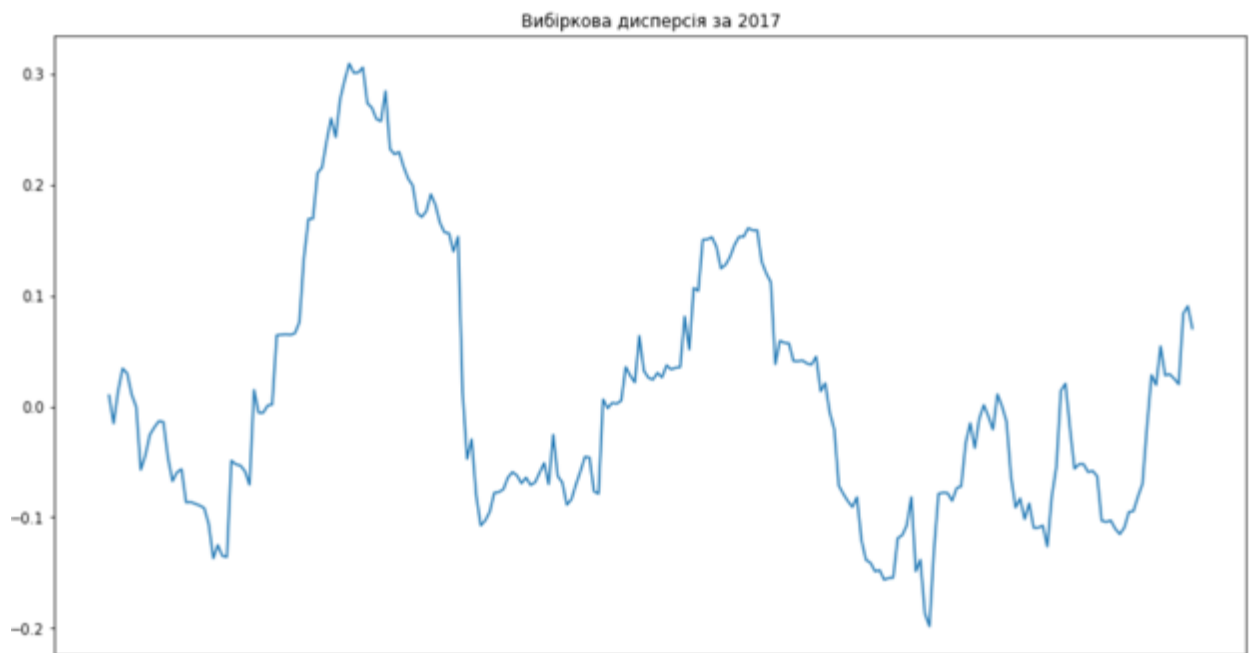


Рисунок 3.16 –Графік вибіркової дисперсії даних за 2017 рік

Обираємо модель для прогнозування значень відносних цін та оцінки можливих втрат серед моделей ARCH, GARCH, EGARCH. Для цього виконуємо моделювання та оцінюємо якість отриманих результатів з використанням критеріїв MSE, R^2 , Байєсівського інформаційного критерію та інформаційного критерію Акаїке (таблиця 3.7).

Таблиця 3.7 – Критерії адекватності моделей

Назва критерію	Модель ARCH	Модель GARCH	Модель EGARCH
MSE	0.168	0.172	0.159
Інформаційний критерій Акаїке	261.4	271.45	252.7
Байєсівський інформаційний критерій	293.8	321.3	283.4
Коефіцієнт детермінації R^2	0.97	0.965	0.974

Визначено, що модель EGARCH має кращі значення критеріїв, тому обираємо її для прогнозування. Розраховані шляхом динамічного прогнозу дані та можливі ризики з використанням моделі EGARCH зображені на рисунку 3.17. Відображено можливі ризики, розраховані з рівнем довіри 95% та 99%. На рисунку 3.17 також зображено фактичні дані за той же період часу.

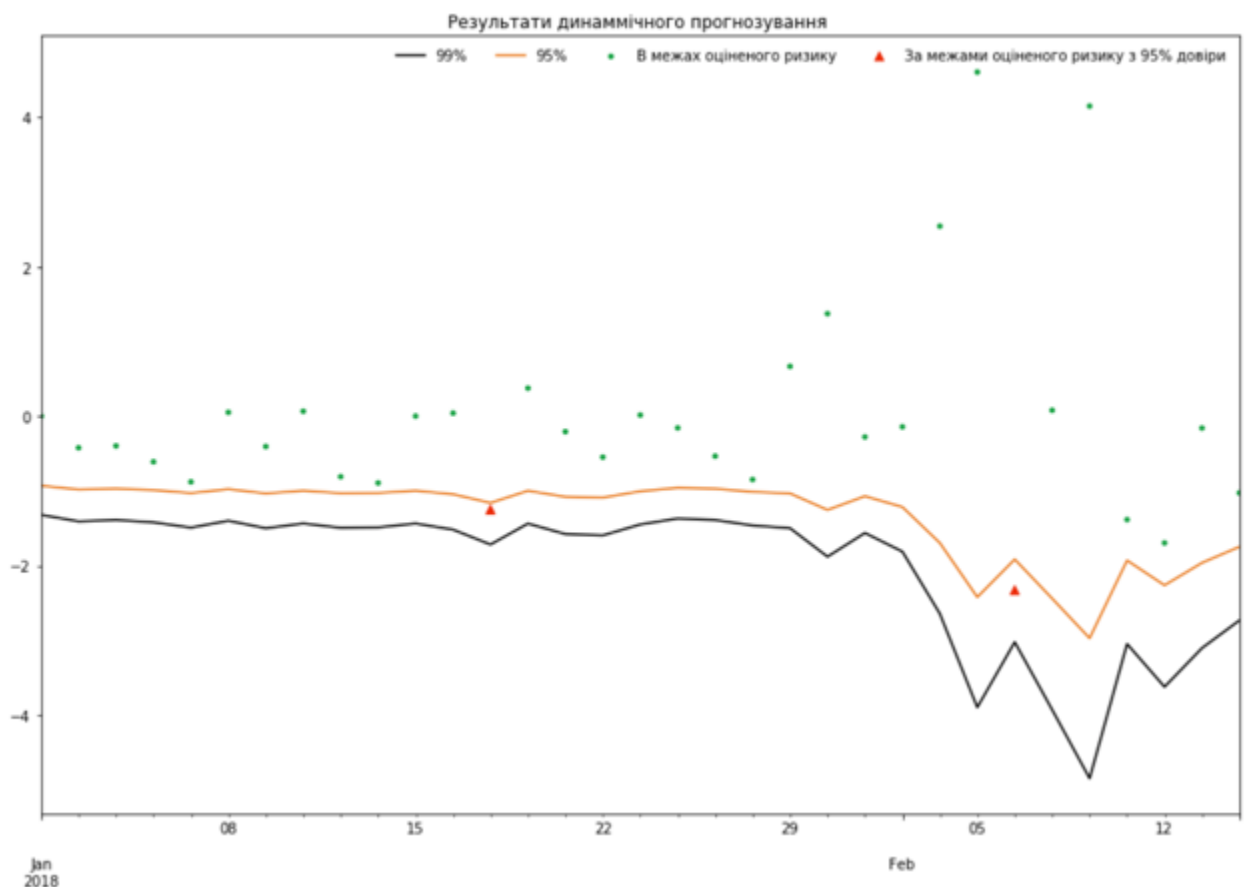


Рисунок 3.17 – Результати динамічного прогнозування моделлю EGARCH

3.4 Висновки до розділу

Для даних з наявним трендом побудована модель ARIMA. Визначено, що модель дає прийнятні результати з урахуванням тренду в даних та слабкої взаємної залежності в ряді різниць.

Для моделювання гетероскедастичних процесів було використано моделі ARCH, GARCH, EGARCH. Визначено, що для обчислювального експерименту з обраним набором даних модель EGARCH найбільш ефективна.

Фактичні дані не виходять за межі прогнозних даних, з урахуванням ризиків, значення яких близькі до них, як для процесів, що мають тренд, так і для тих, що мають ознаки гетероскедастичності.

Розділ 4 ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Постановка задачі

Розробити програмне забезпечення для моделювання та роботи з часовими рядами. Програма реалізується мовою програмування Python та не має обмежень щодо середовища використання (операційна система, апаратне забезпечення). Середовище розробки Jupyter Notebook та PyCharm (Community Edition).

Задля вибору оптимального варіанту реалізації модулю як в економічному плані, так і в плані характеристик буде використаний апарат функціонально-вартісного аналізу.

4.2 Обґрунтування функцій програмного продукту

З наявних цілей програмного продукту можна виділити такі основні функції:

- F₁ – обробка вхідних даних
- F₂ – демонстрація результатів
- F₃ – інтерфейс користувача
- F₄ – застосування моделі для прогнозування

Для кожної з наведених функцій можна виділити декілька варіантів відповідей:

Для F₁:

- а) зчитування даних з файлу
- б) взяття даних із онлайн-сервісу

Для F₂:

- а) виведення результатів на екрані
- б) збереження результатів у файл

Для F₃:

- а) використовувати стандартну бібліотеку PySimpleGUI
- б) використовувати стандартну бібліотеку PyQt

Для F₄:

- а) застосування моделі ARIMA
- б) застосування нейронних мереж
- в) застосування узагальнених моделей

У морфологічній карті системи зображені варіанти комбінацій основних функцій (рисунок 4.1).

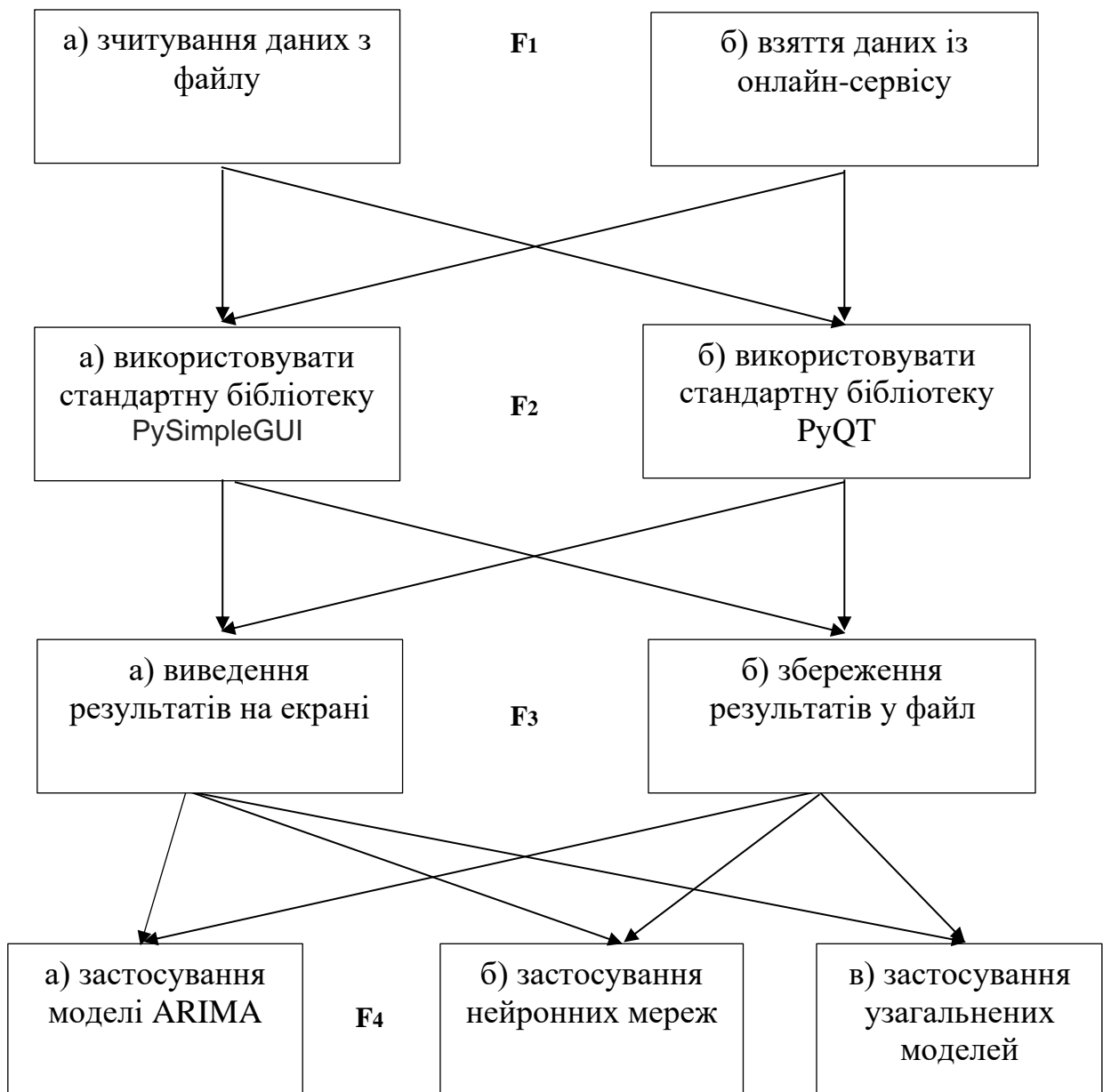


Рисунок 4.1 – Морфологічна карта

За допомогою морфологічної карти побудовано позитивно-негативну матрицю варіантів основних функцій (таблиця 4.1).

Таблиця 4.1 - Позитивно-негативна матриця

Основні функції	Варіанти реалізації	Переваги	Недоліки
F ₁	а	Проста обробка даних, постійний доступ.	Потребується попередня робота з даними.
	б	Отримання даних в реальному часі	Висока складність опрацювання, залежність від онлайн-порталу.
F ₂	а	Менша залежність від сторонніх бібліотек.	Відсутній дизайнер форм, складний у створенні.
	б	Стабільний інструмент з великою кількістю можливостей.	Можливі проблеми пов'язані з мовою C++, що є основою PyQT.
F ₃	а	Результати можна одразу бачити та робити висновки.	Відсутня історія попередніх тестів.
	б	Результати можна зберігати та використовувати для порівняння з наступними.	Можливі проблеми з декодуванням символів програми.
F ₄	а	Наочна та нескладна в реалізації модель, що не потребує великих ресурсів.	Модель може не враховувати фактори, що наявні в обраних за темою даних.
	б	Зазвичай дає хороший результат.	Відсутній наочний контроль за моделлю, складність корегування та реалізації.
	в	Враховує змінну дисперсію, яка притаманна обраний темі.	Оптимізація параметрів може потребувати часу.

Проаналізувавши позитивно-негативну матриці можна дійти до висновку, що при розробці програмного продукту частину варіантів основних функцій можна не розглядати, бо їх використання не сприяє виконанню поставлених перед програмним продуктом задач. Такими є варіанти реалізації F₁-б), F₄-б), F₂-а) та F₃-б). Отже залишились такі варіанти реалізації:

1. F₁-а) – F₂-б) – F₃-а) – F₄-а)
2. F₁-а) – F₂-б) – F₃-а) – F₄-в)

Для оцінювання описаних функцій запропонована система параметрів. Опишемо цю систему.

4.3 Обґрунтування системи параметрів програмного продукту

Визначимо основні параметри, щоб охарактеризувати продукт.

X1 – швидкодія обробки мовою програмування; X2 – часова складність робочих модулів (тривалість роботи програми, або окремих блоків); X3 – ймовірний об'єм коду (вказує на кількість строк коду); X4 – точність результатів (характеризує точність розв'язків, обчислюючи вектор нев'язки)

4.4 Кількісна оцінка параметрів

Гірші, середні та кращі показники параметрів вибираються на основі вимог замовника та умов експлуатації програмного продукту (Таблиця 4.2).

Таблиця 4.2 – Основні параметри дослідження

Умовні позначення	Одиниці виміру	Значення параметрів		
		гірші	середні	кращі
X1	Оп/мс	2000	11000	19000
X2	Мс	2500	1000	750
X3	Кількість строк коду	1200	1000	700
X4	Число	1e-3	1e-4	1e-5

На основі таблиці 4.2 будуються графічні характеристики параметрів (рисунок 4.2 – 4.5).

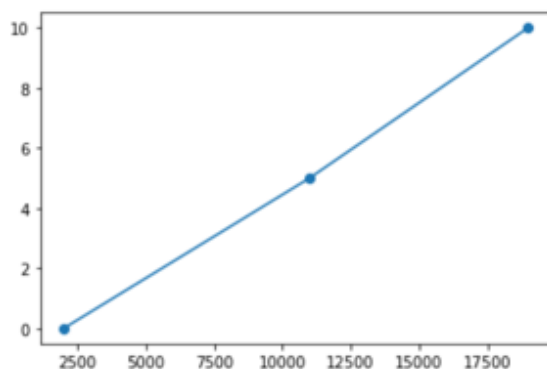


Рисунок 4.2 – Параметр X1

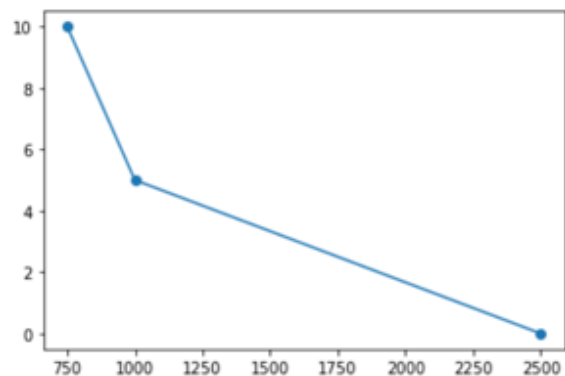


Рисунок 4.3 – Параметр X2

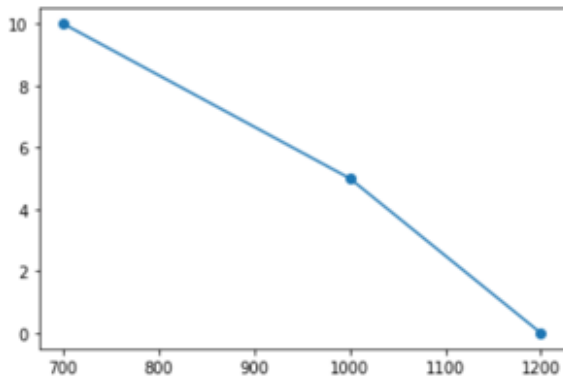


Рисунок 4.4 – Параметр X3

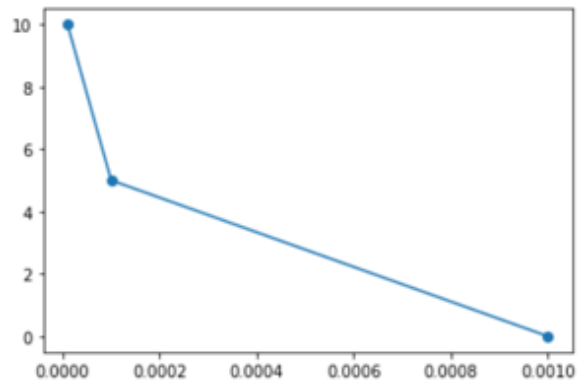


Рисунок 4.5 – Параметр X4

4.5 Аналіз експертного оцінювання параметрів

На основі ґрунтовного обговорення й аналізу група експертів визначає значимість кожного параметра для конкретно поставленої цілі – розробка програмного продукту.

Значимість кожного параметра визначається методом попарного порівняння. Оцінку проводить експертна комісія із 7 людей.

Результати експертного ранжування наведені у таблиці 4.3.

Таблиця 4.3 – Результати експертного ранжування параметрів

Позначення параметра	Ранг параметра за оцінкою експерта							Сума рангів R_i	Відхилення Δ_i	Δ_i^2
	1	2	3	4	5	6	7			
X1	3	4	4	3	4	4	3	25	7.5	56.25
X2	4	3	3	4	2	3	4	23	5.5	30.25
X3	2	2	2	2	3	1	2	14	-3.5	12.25
X4	1	1	1	1	1	2	1	8	-9.5	90.25
Разом	10	10	10	10	10	10	10	70	0	189

Для перевірки ступеню достовірності експертних оцінок, визначимо наступні параметри:

- а) сума рангів кожного з параметрів і загальна сума рангів;
- б) середня сума рангів;
- в) відхилення суми рангів кожного параметра від середньої суми рангів;
- г) загальна сума квадратів відхилення.

Обчислимо коефіцієнт узгодженості:

$$W = \frac{12S}{N^2(n^3 - n)} = \frac{12 * 189}{7^2(4^3 - 4)} \approx 0.77 > 0.67$$

Експертне ранжування достовірне, виходячи з отриманої нерівності.

Проведемо попарне порівняння всіх параметрів і результати занесемо у таблицю 4.4. Зазначимо, що за найбільший ранг приймаємо 4, найменший - 1.

Таблиця 4.4 – Попарне порівняння параметрів

Параметри	Експерти							Кінцева оцінка	Числове значення
	1	2	3	4	5	6	7		
X1 та X2	<	>	>	<	>	>	<	>	1.5
X1 та X3	>	>	>	>	>	>	>	>	1.5
X1 та X4	>	>	>	>	>	>	>	>	1.5
X2 та X3	>	>	>	>	<	>	>	>	1.5
X2 та X4	>	>	>	>	>	>	>	>	1.5
X3 та X4	>	>	>	>	>	<	>	>	1.5

4.6 Аналіз рівня якості варіантів реалізації функцій

Для кожного параметра зробимо розрахунок вагомості K_{vi} за наступною формулою:

$$K_{vi} = \frac{b_i}{\sum_{j=1}^n b_j},$$

де $b_i = \sum_{j=1}^N a_{ij}$.

Відносні оцінки розраховуються декілька разів доти, поки наступні значення не будуть незначно відрізнятися від попередніх (менше 2%). На другому і наступних кроках відносні оцінки розраховуються за формулою:

$$K_{vi} = \frac{b'_i}{\sum_{j=1}^n b'_j},$$

де $b'_i = \sum_{j=1}^N a_{ij} b_j$.

Розрахунок вагомості внесемо до таблиці 4.5.

Таблиця 4.5 – Розрахунок вагомості параметрів

Xi	Xj				Перша ітерація		Друга ітерація		Третя ітерація	
					b_i	K_{vi}	b_i^1	K_{vi}^1	b_i^2	K_{vi}^2
X1	1	1.5	1.5	1.5	5.5	0.3437	21.25	0.3602	77.875	0.3605
X2	0.5	1	1.5	1.5	4.5	0.2813	16.25	0.2754	59.125	0.2739
X3	0.5	0.5	1	1.5	3.5	0.2188	12.25	0.2076	44.875	0.2078
X4	0.5	0.5	0.5	1	2.5	0.156	9.25	0.1568	34.125	0.1578
Всього					16	1	59	1	216	1

Як видно з таблиці, різниця значень коефіцієнтів вагомості не перевищує 2%, тому більшої кількості ітерацій не потрібно.

4.7 Аналіз рівня якості варіантів реалізації функцій

Коефіцієнт технічного рівня для кожного варіанта реалізації програмного продукту розраховується так (таблиця 4.6):

$$K_K(j) = \sum_{i=1}^n K_{vi,j} B_{i,j}$$

Таблиця 4.6 – Розрахунок показників рівня якості варіантів реалізації основних функцій програмного продукту

Основні функції	Варіант реалізації	Параметри	Абсолютне значення параметра	Бальна оцінка параметра	Коеф. вагомості параметра	Коеф. рівня якості
F1	а)	X1	11000	5	0.3605	1.8025
F2	б)	X2	850	7	0.2739	1.9173
F3	а)	X3	700	10	0.2078	2.078
F4	а)	X4	5e-4	3	0.1578	0.4734
	в)	X4	1e-5	7	0.1578	1.1046

Отже отримали:

$$K_K(1) = 1.8025 + 1.9173 + 2.078 + 0.4734 = 6.2712$$

$$K_K(2) = 1.8025 + 1.9173 + 2.078 + 1.1046 = 6.9024$$

Таким чином, другий варіант, що передбачає використання узагальнених моделей дає більший результат. Тому віддаємо йому перевагу.

4.8 Економічний аналіз варіантів розробки програмного продукту

Для оцінки трудомісткості розробки спочатку проведемо розрахунок трудомісткості. Обидва варіанти мають наступні основні завдання:

- 1.1 Розробка алгоритмічної бази та методів розв'язку (ARIMA)
- 1.2 Розробка алгоритмічної бази та методів розв'язку (узагальнені моделі)
2. Розробка програмного продукту;
3. Виведення інформації на екран;

Для завдання 1, при реалізації варіанту 1 (алгоритм групи складності 3, ступінь новизни А, вид використаної інформації — НДІ) $T_P=27$ людино-днів, $K_{\Pi}=1.26$ для НДІ, $K_{СК}=1$, $K_{СТ.М}=1.3$.

$$T_{11} = 27 * 1.26 * 1.3 = 44.226 \text{ людино-днів}$$

Для завдання 1, при реалізації варіанту 2 (алгоритм групи складності 3, ступінь новизни А, вид використаної інформації — НДІ) $T_P=27$, $K_{\Pi}=1.26$ для НДІ, $K_{СК}=1$, $K_{СТ.М}=1.4$.

$$T_{12} = 27 * 1.26 * 1.4 = 47.628 \text{ людино-днів}$$

Для другого завдання (алгоритм групи складності 2, ступінь новизни А, вид використаної інформації — НДІ) $T_P=36$, $K_{\Pi}=1.51$ для НДІ, $K_{СК}=1$, $K_{СТ.М}=1.6$.

$$T_2 = 36 * 1.51 * 1.6 = 86.976 \text{ людино-днів}$$

Для завдання 3 (алгоритм групи складності 2, ступінь новизни Б, вид використаної інформації НДІ) $T_P=27$ людино-днів, $K_{\Pi}=1.08$, $K_{СК}=1$, $K_{СТ}=0.8$, $K_M=1$.

$$T_3 = 27 * 1.08 * 0.8 = 23.328 \text{ людино-днів}$$

Загальна кількість людино-днів:

$$T_1 = 44.226 + 86.976 + 23.328 = 154.53 \text{ людино-днів}$$

$$T_2 = 47.628 + 86.976 + 23.328 = 157.932 \text{ людино-днів}$$

В розробці та проведенні дослідження приймають участь 1 програміст з окладом 18 000 грн та один математик-теоретик з окладом 10 000.

Зарплата розробників становить погодинно:

$$\frac{18000+10000}{2*22*8} = 79.54 \text{ грн.}$$

Зарплата поваріантно:

$$C_1 = 79.54 * 154.53 * 8 = 98\,330 \text{ грн.}$$

$$C_2 = 79.54 * 157.932 * 8 = 100\,495 \text{ грн.}$$

Відрахування на соціальний внесок становить 22%:

$$C_1^v = 0.22 * 98\,330 = 21\,632 \text{ грн.}$$

$$C_2^v = 0.22 * 100\,495 = 22\,108 \text{ грн.}$$

Визначаємо витрати на оплату однієї машино-години, враховуючи заробітну плату програміста в розмірі 18000 грн з коефіцієнтом зайнятості 0.4:

$$C_{\Gamma} = 12 * M * K_3 = 12 * 18000 * 0.4 = 86\,400 \text{ грн.}$$

З урахуванням додаткової заробітної плати:

$$C_{3П} = C_{\Gamma} * (1 + K_3) = 86\,400 * (1 + 0.2) = 103\,680 \text{ грн.}$$

Відрахування на соціальний внесок:

$$C_{ВІД} = C_{3П} * 0.22 = 103\,680 * 0.22 = 22\,809 \text{ грн.}$$

Амортизаційні відрахування розраховуємо при амортизації 25% та вартості ЕОМ – 22000 грн.

$$C_A = K_{TM} * K_A * Ц_{ПР} = 1.15 * 0.25 * 22000 = 6325 \text{ грн.}$$

Витрати на ремонт та профілактику можна підрахувати:

$$C_P = K_{TM} * Ц_{ПР} * K_P = 1.15 * 22000 * 0.05 = 1265 \text{ грн.}$$

Ефективний годинний фонд часу ПК за рік розраховуємо за формулою:

$$T_{EF} = (D_K - D_B - D_C - D_P) * t_3 * K_B = (365 - 104 - 11 - 16) * 8 * 0.9 = 1684.8 \text{ годин}$$

Тепер рахуємо витрати на оплату електроенергії (з урахуванням ПДВ):

$$C_{EL} = T_{EF} * N_C * K_3 * Ц_{ЕН} = 1684.8 * 0.3 * 0.6 * (1.46255 * 1.2) = 532.25 \text{ грн.}$$

Накладні витрати рахуються наступним чином:

$$C_H = Ц_{ПР} * 0.67 = 22000 * 0.67 = 14740 \text{ грн.}$$

Річні експлуатаційні витрати:

$$C_{EKC} = 103\,680 + 22\,809 + 6325 + 1265 + 532.25 + 14740 = 149\,351.25 \text{ грн.}$$

Собівартість однієї машино-години ЕОМ становитиме:

$$C_{M-G} = C_{EKC} / T_{EF} = 149\,351.25 / 1684.8 = 88.64 \text{ грн/год}$$

Витрати на оплату машинного часу складають в залежності від варіанту:

$$C_M^1 = 88.64 * 154.53 * 8 = 109\,580.31 \text{ грн.}$$

$$C_M^2 = 88.64 * 157.932 * 8 = 111\,992.74 \text{ грн.}$$

Накладні витрати складають 67% від заробітної плати:

$$C_H^1 = 0.67 * 109\,580.31 = 73\,418.80 \text{ грн.}$$

$$C_H^2 = 0.67 * 111\,992.74 = 75\,035.13 \text{ грн.}$$

Таким чином, вартість розробки ПП та проведення дослідів складає:

$$C_1 = 98\,330 + 21\,632 + 109\,580.31 + 73418.80 = 302\,961.11 \text{ грн.}$$

$$C_2 = 100\,495 + 22\,108 + 111\,992.74 + 75035.13 = 309\,630.67 \text{ грн.}$$

Проведемо розрахунок техніко-економічного рівня:

$$K_{\text{TEP1}} = 6.2712 / 302\,961.11 = 2.07 \cdot 10^{-5},$$

$$K_{\text{TEP2}} = 6.9024 / 309\,630.67 = 2.23 \cdot 10^{-5}.$$

Отже найбільш ефективним є другий варіант з коефіцієнтом техніко-економічного рівня $2.23 \cdot 10^{-5}$.

4.9 Висновки до розділу

Було проведено повний функціонально-вартісний аналіз програмного продукту, який було розроблено в рамках дипломного проекту. За допомогою технічного дослідження було виділено 2 можливих варіанти реалізації ПП. Після їх порівняння з економічної точки зору, було визначено, що 2-ий варіант реалізації є більш доцільним з техніко-економічним рівнем $2.23 \cdot 10^{-5}$. Цей варіант передбачає:

1. Зчитування даних з файлу
2. Застосування узагальнених моделей
3. Виведення результатів на екрані
4. Використовувати стандартну бібліотеку PyQt

Даний варіант виконання програмного продукту надає користувачу зручний інтерфейс, досить потужний функціонал і прийнятну швидкодію.

ВИСНОВКИ

В роботі проведено огляд сучасних підходів до моделювання нестационарних процесів та процесів зі змінною в часі дисперсією. Для динамічного оцінювання можливих втрат в нестационарних фінансово-економічних процесах обрано використання прогнозів волатильності як міри ризику. На основі проведеного огляду існуючих моделей прогнозування часових рядів з урахуванням змінної волатильності виконано їх порівняння, проаналізовано переваги та недоліки з урахуванням складності моделей та вимог до точності результатів.

Для аналізу адекватності побудованих моделей та оцінки якості прогнозування визначено основні критерії, такі як: середньоквадратична похибка, сума квадратів похибок моделі, статистика Дарбіна-Уотсона, коефіцієнт детермінації R^2 , інформаційний критерій Акаїке, критерій Маллоуза та коефіцієнт Тейла.

За результатами функціонально-вартісного аналізу обрано оптимальний варіант реалізації програмного продукту.

Розроблено програмне забезпечення для побудови авторегресійних моделей, таких як ARIMA, ARCH, GARCH, EGARCH, і визначення нестационарності та гетероскедастичності поданих даних на основі статистичних тестів. Передбачено можливість перевірки адекватності створених моделей. В результаті використання прогнозів волатильності реалізовано динамічну оцінку ризиків з використанням значення VaR.

За підсумками проведених обчислювальних експериментів визначено, що побудовані в рамках дослідження моделі забезпечують бажаний результат із заданим рівнем довіри, розрахована оцінка ризиків підтверджує здатність моделей до ефективного прогнозування при нестабільній ситуації на ринку.

РЕКОМЕНДАЦІЇ

З метою підвищення точності результатів в процесах з високим рівнем волатильності для дослідження рекомендується використовувати моделі, що враховують вплив фінансового левериджу.

З урахуванням зростаючих обчислювальних можливостей комп'ютерних систем, в якості більш сучасного та потужного інструменту аналізу та прогнозування пропонується використовувати нейронні мережі.

Розроблене програмне забезпечення може бути використано для динамічного оцінювання фінансових ризиків компаній з урахуванням нестабільності ринку, яка в останні роки особливо відчувається у всьому світі, і бути фактором для прийняття рішень щодо інвестицій, надаючи достатньо точні розрахунки можливих втрат.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Словник української мови: у 11 т. / АН УРСР. Інститут мовознавства; за ред. І.К. Білодіда. К.: Наукова думка, 1977. Т. 8. С. 153.
2. Box G.E.P., Jenkins G.M., Reinsel G.C., Ljung G.M. Time Series Analysis: Forecasting and Control. Wiley, 2015. 712 p.
3. Finanzielles Risiko. *Capital Com*. URL: <https://capital.com/de/finanzielles-risiko-definition>.
4. Enterprise risk management. Financial & Credit Risks. *Dun & Bradstreet*. URL: <https://www.dnb.com/resources/finance-credit-risk.html>.
5. Шапкин А.С., Шапкин В.А. Теория риска и моделирование рисков ситудаций. М.: Дашков и Ко, 2005. 544 с.
6. Karen A. Horcher. ESSENTIALS of Financial Risk Management. Wiley, 2011. 274 p.
7. Jimmy Skoglund, Wei Chen. Financial Risk Management: Applications in Market, Credit, Asset and Liability Management and Firmwide Risk. Wiley, 2015, 557 p.
8. Дрейпер Н., Смит Г. Прикладной регрессионный анализ: у 2 т. М.: Финансы и статистика, 1986. Т. 2. 366 с.
9. Бідюк П.І., Половцев О.В. Аналіз та моделювання економічних процесів перехідного періоду. К.: НТУУ КПІ, 1999. 230 с.
10. Бідюк П.І., Романенко В.Д., Тимошук О.Л. Аналіз часових рядів: навч. посіб. К.: Політехніка, 2010. 317 с.
11. Walid, Subanar, Dedi Rosadi, Suhartono. Fractional integrated recurrent neural network (FIRNN) for forecasting of time series data in electricity load in Java-Bali. *Contemporary Engineering Sciences*. 2015. Vol. 8, No. 32, P. 1535–1550 DOI:10.12988/ces.2015.510283
URL: https://pdfs.semanticscholar.org/c9bd/04a003a9c287acdcec3818c9a01ba2115dad.pdf?_ga=2.41225888.38804288.1591023131-1203284083.1591023131.

12. Introduction to Generalized Linear Models. *The Pennsylvania State University*.
URL: <https://online.stat.psu.edu/stat504/node/216/>.
13. An Introduction to EViews. *EViews*. URL:
<https://www.eviews.com/Discovering/whatisreviews.html>.
14. A. Aljandali, M. Tatahi. Economic and Financial Modelling with EViews: A Guide for Students and Professionals. Springer International Publishing, 2018. 293 p.
15. Дробыш И.И. Статистические модели прогнозирования цен на оптовых рынках электроэнергии: российский и зарубежный опыт / И.И. Дробыш // Вестник ЮУрГУ. Серия «Экономика и менеджмент». 2017. Т. 11, No 3. С. 41–54. DOI: 10.14529/em170306.
16. Sarmila Banerjee, Anjan Chakrabarti. Development and Sustainability: India in a Global Perspective. Springer, 2013. 617 p.
17. G.S. Maddala, In-Moo Kim. Unit Roots, Cointegration, and Structural Change. Cambridge University Press, 2004. 505 p.
18. Roberto Pedace. Econometrics For Dummies. Wiley, 2013. 368 с.
19. Евсеев Е.А. Эконометрика: учебное пособие для академического бакалавриата / Е. А. Евсеев, В. М. Буре. 2-е изд., испр. и доп. М.: Юрайт, 2017. 186 с.
20. Lecture 15. Model Building: Selection Criteria, 2011. P. 15–17. URL:
https://www.stat.purdue.edu/~ghobbs/STAT_512/Lecture_Notes/Regression/Topic_15.pdf.

ДОДАТОК А ПРОГРАМНИЙ КОД

```

import matplotlib.pyplot as plt
import numpy as np
import statsmodels
from statsmodels.tsa.stattools import adfuller as DickeyFuller
from statsmodels.tsa.stattools import pacf, acf
import itertools
import copy
import statistics
from functools import reduce
import pandas as pd
import datetime as dt
import pandas_datareader.data as web
from arch import arch_model

def MA(series, window = 5):
    series = np.cumsum(series)
    series[window:] = series[window:] - series[:-window]
    return series[window - 1:] / window

def MSE(series, predict):
    return np.sum((series-predict)**2)/len(series)

def SSE(series, predict):
    return np.sum((series-predict)**2)

def r2(predictions, y):
    l = len(predictions)
    avg = reduce(lambda v,w: v + w, y)/l
    dispersion = 0
    testDispersion = 0
    for i in range(l):
        dispersion += (y[i] - predictions[i])**2
        testDispersion += (y[i] - avg)**2
    r2 = (testDispersion/(testDispersion + dispersion)) ** 2
    return r2

def DW(y, pred):
    e = list(np.array(y) - np.array(pred))
    sq = sum(list(map(lambda x: x**2, e)))
    diff = np.array(e[1:]) - np.array(e[:-1])
    diff = diff @ diff
    val = diff/sq
    return val

def variance(Y, length = 0):
    if length == 0:
        length = len(Y)
    mean = 0
    variance = 0
    for i in range(len(Y)-length, len(Y)):

```

```

    mean += Y[i]
    mean /= length
    for i in range(len(Y)-length,len(Y)):
        variance += (Y[i] - mean)**2
    variance /= (- 1 + length)
    return variance

def k(interval):
    if interval == 0.95:
        return 1.65
    elif interval == 0.99:
        return 2.33

    else:
        return 1.5

def VaR(value, var, alpha = 0.05, verbose = False):
    new_value = value*(1-np.exp(-(k(1-alpha)*var)))
    if verbose:
        print(new_value)
    return k(1-alpha)*var

def define_orders(series, horizon_AR=10,horizon_MA = 10, limit=0.6, verbose = False):
    p = 0
    q = 0
    for i in range(horizon_MA - 1, 0, -1):
        if verbose:
            print('acf ', i, ' ', abs(r(series,i)))
            print('pacf ', i, ' ', abs(pacf_my(series,i)))

        if abs(r(series,i)) > limit and q == 0:
            q = i

    for i in range(horizon_AR - 1, 0, -1):
        if abs(pacf_my(series,i)) > limit and p == 0:
            p = i

    return p, q

def r(Y, k):
    mean = 0
    variance = 0
    r = 0
    for i in range(len(Y)):
        mean += Y[i]
    mean /= len(Y)

    #(N-1) * variance
    for i in range(len(Y)):
        variance += (Y[i] - mean)**2
    variance /= (len(Y) - 1)

```

```

for i in range(k, len(Y)):
    r += (Y[i] - mean)*(Y[i - k] - mean)
r /= variance
return r / (len(Y) - 1)
class PACF(object):
    def __init__(self, time_s):
        self.time_s = np.array(time_s)
        self.pacf_results = {}
        self.time_s_mean = self.time_s.mean()
        self.time_s_var = self.count_var()

    def count_var(self):
        return ((self.time_s - self.time_s_mean)**2).sum() / (self.time_s.shape[0] - 1)

    def count_r(self, s):
        to_div = sum( (self.time_s[i] - self.time_s_mean) * (self.time_s[i-s] -
self.time_s_mean) for i in range(s, len(self.time_s)))
        divider = (self.time_s_var)*(len(self.time_s) - 1)
        return to_div / divider

    def F(self, k,j):

        if (k,j) in self.pacf_results.keys():
            return self.pacf_results[(k,j)]

        if k == j == 1 :
            f = self.count_r(1)
        else:
            to_div = self.count_r(k) - sum(self.F(k - 1, i) * self.count_r(k - i) for i in range(1, k))
            divider = 1 - sum(self.F(k - 1, i) * self.count_r(i) for i in range(1, k))

            f = to_div/divider

        if k == j :
            self.pacf_results[(k, j)] = f

```

```

        return self.pacf_results[(k, j)]
    else:
        self.pacf_results[(k, j)] = self.F(k - 1, j) - f * self.F(k - 1, k - j)
        return self.pacf_results[(k, j)]

def simmetric_F(self, k):
    return self.F(k, k)

def pacf_my(Y, s):
    values = {}

    def F(k, j):
        if((k, j) in values.keys()):
            return values[(k, j)]
        else:
            if(k == 1 and j == 1):
                f = r(Y, 1)
            else:
                numerator = 0
                denominator = 0

                for i in range(1, k):
                    numerator -= F(k - 1, i) * r(Y, k - i)
                    denominator -= F(k - 1, i) * r(Y, i)
                numerator += r(Y, k)
                denominator += 1

                f = numerator/denominator

            if(k == j):
                values[(k, j)] = f
                return values[(k, j)]
            else:
                values[(k, j)] = F(k - 1, j) - f * F(k - 1, k - j)
                return values[(k, j)]

    def G(k):
        if(k == 1):
            return r(Y, 1)
        else:
            numerator = 0
            denominator = 0

            for i in range(1, k):
                numerator -= F(k - 1, i) * r(Y, k - i)
                denominator -= F(k - 1, i) * r(Y, i)
            numerator += r(Y, k)
            denominator += 1

```

```

        return numenator/denominator
    return G(s)

```

```

def ARIMA_model(series,procent = '5%', horizon_ar = 15, horizon_ma = 10, pacf_limit=
0.4, window = 5, to_plot = True):
    current_series = series
    counter = 0
    while True:
        testDF = DickeyFuller(current_series)
        if testDF[0] > testDF[4][procent]:#гипотеза не прошла
            diff_series = current_series[1:] - current_series[:-1]
            current_series = diff_series
            counter += 1
        else:
            break
    if counter == 0:
        trend_predict = np.zeros(len(series))
        trend_coefs = []
    else:
        trend_data = np.array([np.power(np.arange(len(series)),i) for i in range(counter+1)]).T
        trend_coefs = np.linalg.lstsq(trend_data, series)[0]
        trend_predict = trend_data@trend_coefs.T
    diff = series - trend_predict
    p, q = define_orders(diff,horizon_AR = horizon_ar, horizon_MA = horizon_ma, limit =
pacf_limit)
    print('Trend order: ', counter)
    print('Order of AR: ', p)
    print('Order of MA', q)
    border_AR = max(p,q + window)
    border_MA = border_AR - window + 1
    data_AR = np.array([diff[border_AR - i:-i-1] for i in range(p)]).T
    data_MA = np.array([MA(diff[border_MA - i:-i-1], window) for i in range(q)]).T
    if p == 0 and q == 0:
        print('p and q = 0')
        return trend_coefs, 0, 0
    if p == 0:
        data = data_MA
    elif q == 0:
        data = data_AR
    else:
        data = np.hstack((data_AR,data_MA))
    coefs = np.linalg.lstsq(data,diff[border_AR + 1:])[0]
    if p == 0:
        coefs_ma = coefs[p:]
        coefs_ar = []
    elif q == 0:
        coefs_ma = []
        coefs_ar = coefs[:p]
    else:
        coefs_ar = coefs[:p]
        coefs_ma = coefs[p:]

```



```

diff_predict = data@coefs.T
result = diff_predict + trend_predict[border_AR + 1:]
if not to_plot:
    plt.figure(figsize=(20,8))
    plt.plot(diff_predict)
    plt.plot(diff[border_AR + 1:])
    plt.title('Approximation with ARIMA(' + str(p) + ',' + str(counter) + ',' + str(q) + ')')
    plt.legend(['Approximation', 'Real values'])

if to_plot:
    plt.figure(figsize=(20,8))
    plt.plot(result)
    plt.plot(series[border_AR + 1:])
    plt.title('Approximation with ARIMA(' + str(p) + ',' + str(counter) + ',' + str(q) + ')')
    plt.legend(['Approximation', 'Real values'])

stats = { }
stats['MSE'] = MSE(series[border_AR + 1:],result)
stats['Mean value'] = np.sum(series[border_AR + 1:])/len(series[border_AR + 1:])
stats['Median value'] = statistics.median(map(float, series[border_AR + 1:]))
stats['Akaike criterion'] = 2*(p+q)-2*np.log(SSE(series[border_AR + 1:],result))
stats['R^2'] = r2(result, series[border_AR + 1:])
stats['Durbin-Watson'] = DW(series[border_AR + 1:],result)
return trend_coefs, coefs_ar, coefs_ma, stats

def get_trend(coef_i, dot):
    trend = 0
    for i in range(len(coef_i)):
        trend += dot**i * coef_i[i]
    return trend

def arima_predict(series, coef_ar, coef_i, coef_ma, window = 5):
    p = len(coef_ar)
    q = len(coef_ma)
    i_trend = len(coef_i)
    series_v_trende = copy.copy(series)
    forecast = list(series)
    for i in range(len(series)):
        series_v_trende[i] = series[i] - get_trend(coef_i, i)
    border_AR = max(p,q + window)
    border_MA = border_AR - window + 1
    data_ma = MA(series_v_trende, window)
    predicted = 0
    for i in range(p):
        predicted += coef_ar[i]*series_v_trende[-i - 1]
    for i in range(q):
        predicted += coef_ma[i]*data_ma[-i - 1]

    predicted += get_trend(coef_i,len(series) )
    return predicted

```

```

coef_i, coef_ar, coef_ma, stats = ARIMA_model(series[:-N_predicted], horizon_ar = 11,
horizon_ma = 8, procent = '5%', pacf_limit=0.2, to_plot=True)
N_predicted = 10
n_list_predicted = [N_predicted-i for i in range(N_predicted)]
new_elems = []
predicted_risk = []
for n in n_list_predicted:
    elem = arima_predict(series[:-n], coef_ar, coef_i, coef_ma)
    predicted_risk.append(VaR(100,variance(np.append(series[:-n],elem), 25)**0.5))
    new_elems.append(elem)
N_display = 35
predicted_risk = np.array(predicted_risk)
maximal_risk = np.append(series[-N_display:-N_predicted],series[-N_predicted-1:-1] +
predicted_risk)
minimal_risk = np.append(series[-N_display:-N_predicted],series[-N_predicted-1:-1] -
predicted_risk)
plt.figure(figsize = (15,10))
real_and_forecast = np.append(series[-N_display:-N_predicted],np.array(new_elems))
index = [i for i in range(len(series)-N_display,len(series))]
plt.plot(index,minimal_risk, color = 'black')

plt.plot(index,real_and_forecast)
plt.plot(index,series[-N_display:])

plt.title('Dynamic forecast')
plt.legend(['Forecast', 'Real values'])

def count_y_hat(time_s, k):
    if k == 0:
        raise ValueError("Tu cho tvocrich!")
    return (1/k) * sum(time_s[:k+1])

def count_cond_dispersion(time_s, k):
    if k == 0:
        raise ValueError("Tu cho tvocrich!")
    return (1/k) * sum((time_s[:k+1]-count_y_hat(time_s, k))**2)
def ARCH(series, horizon = 15, pacf_limit = 0.4, window = 5):
    #standart AR to series
    p, q = define_orders(series,horizon=20,limit = pacf_limit)
    if p == 0:
        return 0
    data = np.array([series[p - i - 1:-i-1] for i in range(p)]).T
    coefs = np.linalg.lstsq(data,series[p:])[0]
    predicted_series = data@coefs.T

    #counting errors^2 and predict them with AR
    errors = series[p:] - predicted_series
    squared_errors = errors**2
    error_dispersion = np.sum(squared_errors)/(len(squared_errors)-1)
    p_error, q = define_orders(squared_errors,horizon=20,limit = pacf_limit)
    data_error = np.array([squared_errors[p_error - i - 1:-i-1] for i in range(p_error)]).T
    coefs_error = np.linalg.lstsq(data_error,squared_errors[p_error:])[0]

```

```

predicted_errors = data_error@coefs_error.T

#create new series as series-errors_approximated
new_errors = abs(predicted_errors)**0.5
new_series_AR = predicted_series[p_error:] + new_errors
new_series = series[len(series)-len(new_errors):] - new_errors

#building ARMA to new series
p_new, q = define_orders(new_series,horizon=horizon,limit = pacf_limit, verbose =
False)
border_AR = p_new#max(p_new,q + window)
border_MA = border_AR - window + 1
data_AR = np.array([new_series[border_AR - i:-i-1] for i in range(p_new)]).T
q=0
#data_MA = np.array([MA(new_series[border_MA - i:-i-1], window) for i in range(q)]).T
if p_new == 0 and q == 0:
    print('p_new and q = 0')
    return 0, 0
if p == 0:
    new_data = data_MA
elif q == 0:
    new_data = data_AR
else:
    new_data = np.hstack((data_AR,data_MA))
new_coefs = np.linalg.lstsq(new_data,new_series[border_AR + 1:])[0]

#adding predicted errors to approximation of new series --- result
new_predicted_series = new_data@new_coefs.T + new_errors[border_AR + 1:]

#print("ARCH MSE")
#print(np.sum((new_predicted_series - series[len(series) -
len(new_predicted_series):])**2)/len(new_predicted_series))

plt.figure(figsize = (20,10))
plt.plot(new_predicted_series)
plt.plot(series[len(series) - len(new_predicted_series):])

#plt.plot(series[p:] - predicted_series)
return coefs, coefs_error, new_coefs

def arch_forecast(series,coefficients, window = 5, to_plot = False):
    coefs, coef_error, coef_ar = coefficients
    p = len(coefs)
    p_AR = len(coef_ar)
    p_error = len(coef_error)

    series_predict = np.zeros(len(series)-p)

    for j in range(len(series)-p):

```

```

    for i in range(p):
        series_predict[-j-1] += series[-i - j - 2]*coefs[i]
    errors = (series[p:] - series_predict)**2
    errors_predict = np.zeros(len(series)-p-p_error)

    for j in range(len(series)-p-p_error):
        for i in range(p_error):
            errors_predict[-j-1] += errors[-i-2-j] * coef_error[i]

    new_series = series[p+p_error:] - abs(errors_predict)**0.5

    new_series_predict = np.zeros(len(new_series)-p_AR)
    for j in range(len(new_series)-p_AR):
        for i in range(p_AR):
            new_series_predict[-j-1] += new_series[-i - j - 2]*coef_ar[i]
    if to_plot:
        plt.figure(figsize = (15,8))
        plt.plot(new_series_predict + errors_predict[p_AR:])
        plt.plot(series[len(series)-len(new_series_predict):])
    predicted = 0
    for i in range(p_AR):
        predicted += new_series[-i - j - 1]*coef_ar[i]
    for i in range(p_error):
        predicted += abs(errors[-i-1-j] * coef_error[i])**0.5

    return predicted

from statsmodels.stats.diagnostic import het_white
import statsmodels.formula.api as smf

import statsmodels.api as sm
food = pd.DataFrame(food, columns=['food_exp','income'])
foodexp = food['food_exp']

income = food['income']
formular = 'foodexp ~ income'
#food = np.hstack((foodexp,income))

foodModel = smf.ols(formula = formular, data = food)

fresults = foodModel.fit()

print(fresults.summary())

results = het_white(fresults.resid, fresults.model.exog)
Arch = arch_model(returns[409:-372], vol='ARCH', p=p, dist='skewt')
GJR = arch_model(returns[409:-372], vol='GARCH', p=p, o=1, q=q, dist='skewt')
Garch = arch_model(returns[409:-372], vol='GARCH', p=p, o=0, q=q, dist='skewt')
Egarch = arch_model(returns[409:-372], vol='EGARCH', p=p, q=q, dist='skewt')
GJR_result = GJR.fit(disp='off', last_obs='2017-12-31')
Garch_result = Garch.fit(disp='off', last_obs='2017-12-31')
Arch_result = Arch.fit(disp='off', last_obs='2017-12-31')

```

```

Egarch_result = Egarch.fit(disp='off', last_obs='2017-12-31')
forecasts = Garch_result.forecast(start='2018-1-1')
cond_mean = forecasts.mean['2018':'2018-5-1']
cond_var = forecasts.variance['2018':'2018-5-1']
quantil = Garch.distribution.ppf([0.01, 0.05], Garch_result.params[-2:])
value_at_risk = cond_mean.values + np.sqrt(cond_var).values * quantil[None, :] - 0.5
value_at_risk = pd.DataFrame(value_at_risk, columns=['99%', '95%'],
index=cond_var.index)
plt.figure(figsize = (15,8))
plt.plot(forecasts.variance.dropna()['h.1'][:87])

ax = value_at_risk.plot(legend=False, figsize = (15,10), color = ['k','#ea7a1d'])
xl = ax.set_xlim(value_at_risk.index[0], value_at_risk.index[-1])
rets_2018 = returns['2018':'2018-5-1'].copy()
rets_2017 = returns['2017':'2017-12-31'].copy()
#rets_2018.name = 'S&P 500 Return'
c = []
main_color = '#16a343'
procent5_color = '#ec2906'
for idx in value_at_risk.index:
    if -rets_2018[idx] > value_at_risk.loc[idx, '95%']:
        c.append(main_color)
    elif -rets_2018[idx] < value_at_risk.loc[idx, '99%']:
        c.append('#BB0000')
    else:
        c.append(procent5_color)

c = np.array(c, dtype='object')
labels = {
    '#BB0000': 'За межами оціненого ризику з 99% довіри',
    procent5_color: 'За межами оціненого ризику з 95% довіри',
    main_color: 'В межах оціненого ризику'}

markers = {'#BB0000': 'x', procent5_color: '^', main_color: '.'}
for color in np.unique(c):
    sel = c == color
    ax.scatter(rets_2018.index[sel],
               -rets_2018.loc[sel],
               marker=markers[color],
               c=c[sel],
               label=labels[color])
ax.set_title('Результати динамічного прогнозування ')
leg = ax.legend(frameon=False, ncol=5)

def r(Y, k):
    mean = 0
    variance = 0
    r = 0
    for i in range(len(Y)):
        mean += Y[i]
    mean /= len(Y)

```

```

#(N-1) * variance
for i in range(len(Y)):
    variance += (Y[i] - mean)**2
variance /= (len(Y) - 1)
for i in range(k, len(Y)):
    r += (Y[i] - mean)*(Y[i - k] - mean)
r /= variance
return r / (len(Y) - 1)
variance_check = values_returns[409:-633]
varian = []
#print(r(test[5:15],5))
for i in range(240):
    varian.append(r(variance_check[i:i+40],26))
plt.figure(figsize = (15,8))
plt.title('Вибіркова дисперсія за 2017')
plt.plot(np.array(varian))

```

ДОДАТОК Б ІЛЮСТРАТИВНИЙ МАТЕРІАЛ



НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КИЇВСЬКИЙ
ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО»
Інститут прикладного системного аналізу
Кафедра математичних методів системного аналізу

Дипломна робота на здобуття ступеня бакалавра

Моделі та оцінки можливих втрат на основі прогнозів волатильності

Виконав: студент групи КА-61
Москаленко Володимир
Володимирович

Керівник: професор, д.т.н.
Бідюк Петро Іванович

АКТУАЛЬНІСТЬ ТЕМИ ДОСЛІДЖЕННЯ

- Потреба точного розрахунку прогнозних даних та оцінки можливих втрат у фінансовій та економічній сфері.
- Необхідність описувати нелінійні та нестационарні фінансові процеси
- Оцінка можливих втрат для прийняття рішень

ОБ'ЄКТ, ПРЕДМЕТ І МЕТА ДОСЛІДЖЕННЯ

Об'єкт дослідження:

нелінійні нестационарні процеси в економіці та фінансах, фінансові ризики

Предмет дослідження:

моделі і методи аналізу нелінійних нестационарних процесів в економіці та фінансах, методи аналізу фінансових ризиків

Мета дослідження:

підвищення якості оцінок прогнозів нелінійних нестационарних процесів та їх волатильності, покращення оцінок ринкового ризику завдяки новим побудованим моделям

3

ПОСТАНОВКА ЗАДАЧІ

1. Виконати огляд сучасних математичних моделей для моделювання і прогнозування нестационарних гетероскедастичних фінансово-економічних процесів, методів оцінювання ринкових ризиків.
2. Зібрати необхідні статистичні дані для виконання обчислювальних експериментів.
3. Створити програмний продукт для виконання обчислювальних експериментів: моделювання і прогнозування процесів зі змінною дисперсією, оцінки ризику.
4. Виконати аналіз отриманих результатів і зробити висновки.

4

СТАТИСТИЧНІ ТЕСТИ ТА КРИТЕРІЇ АДЕКВАТНОСТІ МОДЕЛІ

НЕЛІНІЙНИЙ ПРОЦЕС

1. Тест Дікі-Фуллера:

Тест для перевірки стаціонарності процесу, базується на перевірці наявності одиничного кореня

2. Коефіцієнт детермінації: $R^2 = \frac{\sum_{j=1}^n |y_j - \hat{y}_j|^2}{\sum_{j=1}^n |\bar{y}_j - y_j|^2}$

3. Інформаційний критерій Акаїке: $AIC = \frac{2(k-L)}{n}$

$$L = -\frac{2}{n} \left(1 + \ln(2\pi) + \ln\left(\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2\right) \right)$$

5

ПРИКЛАД МОДЕЛЮВАННЯ

ВХІДНІ ДАНІ



Набір даних Brent Oil Prices

День	Ціна (в доларах)
1	83.76
2	82.9
3	81.28
4	81.54
5	79.89
6	77.83
7	76.63
8	75.14
9	74.56
10	76.74
11	75.1
12	74.84
13	73.48
14	73.08
15	70.61
16	70.74
17	83.76
18	83.76
19	82.9
...	...
196	112.87
197	113.08
198	113.72
199	109.39
200	112.54

6

МОДЕЛЬ ARIMA. ПРОЦЕС МОДЕЛЮВАННЯ

Для створення стаціонарного ряду в моделі ARIMA використовуємо різниці часового ряду порядку d . Порядок d вибираємо за критерієм Дікі-Фуллера.

$$\Delta^d Y_t = c + \sum_{i=1}^p \alpha_i \Delta^d Y_{t-i} + \sum_{i=1}^q b_i \varepsilon_{t-i} + \varepsilon_t$$

Для оцінки коефіцієнтів отриманого ряду використовуємо модуль ARMA

Повертаємось до початкової моделі, перевіряємо її адекватність.

Виконуємо прогнозування даних.

7

КРИТЕРІЇ ЯКОСТІ ПОБУДОВАНОЇ МОДЕЛІ

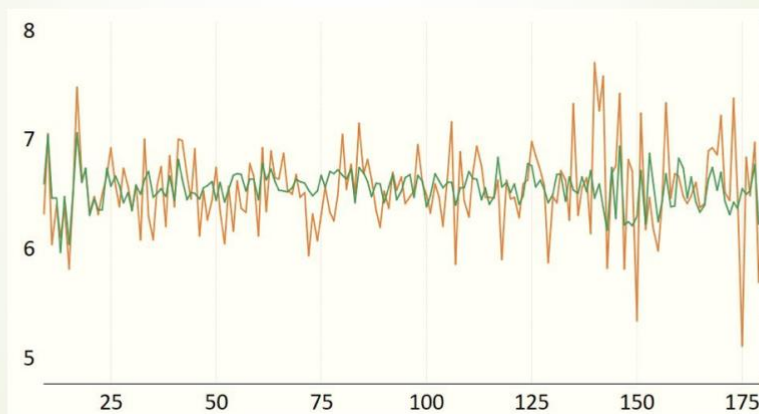
Порівняння значень критеріїв автоматично побудованої моделі ARIMA з використанням Eviews та власної моделі

Назва критерію	Значення програми	Значення EViews
MSE	0.00114	0.0129
Інформаційний критерій Акаїке	3.177	-3.4
Коефіцієнт детермінації R^2	0.93	0.87.
Критерій Дарбіна-Уотсона	1.91	1.88

8

РЕЗУЛЬТАТИ МОДЕЛЮВАННЯ

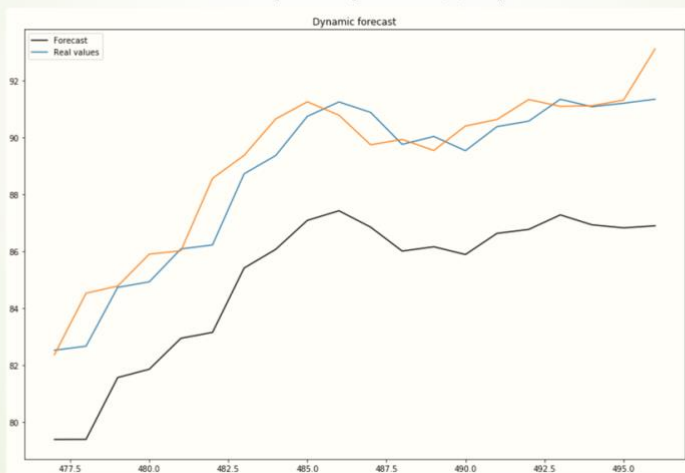
Результат моделювання перших різниць даних цін Brent Oil



9

РЕЗУЛЬТАТИ МОДЕЛЮВАННЯ

Результат динамічного прогнозування цін акцій Brent Oil Prices та оцінки можливих втрат із рівнем довіри 95%

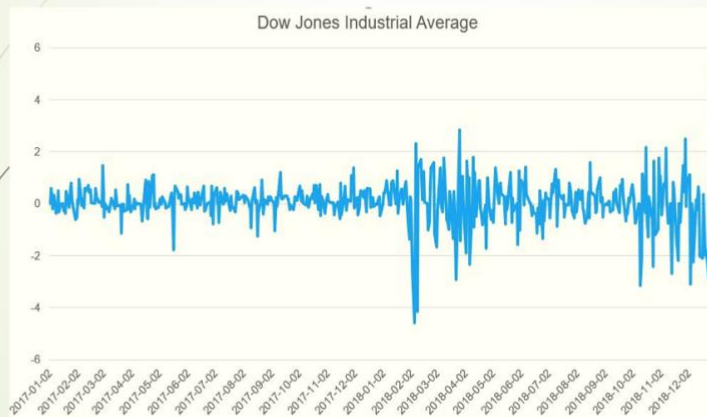


10

ПРИКЛАД МОДЕЛЮВАННЯ

Вхідні дані

Набір даних Dow Jones Industrial Average



День	Ціна (в доларах)
1	-0.0141
2	1.3305
3	0.2164
4	-0.779
5	-0.6015
6	0.6368
7	0.1745
8	1.0041
9	-0.5513
10	0.5763
11	0.134
12	-0.981
13	-0.4214
14	0.3148
15	-1.952
16	0.1316
...	...
258	-0.171
259	0.255
260	-0.476

11

МОДЕЛЬ EGARCH. ПРОЦЕС МОДЕЛЮВАННЯ

Будується модель AP або ARMA, використовуючи показники АКФ та ЧАКФ для поданих даних. Результат моделювання віднімаємо від вхідних даних та таким чином отримуємо ряд залишків $\{\varepsilon(k)\}$

На основі значень ряду залишків та ряду їх квадратів будується модель ARCH/GARCH або їх модифікація.

Рівняння для логарифму умовної дисперсії в EGARCH має вигляд:

$$\log(h(k)) = \beta_0 + \sum_{i=1}^p \beta_i \frac{|\varepsilon(k-i)|}{h(k-i)} + \sum_{i=1}^p \gamma_i \frac{\varepsilon(k-i)}{h(k-i)} + \sum_{i=1}^q \alpha_i \log(h(k-i))$$

Отримавши оцінені значення уточнюємо вихідну модель

$$Y_{\text{уточ}}(k) = Y(k) - \hat{\varepsilon}_{\text{оц}}(k)$$

12

КРИТЕРІЇ ЯКОСТІ ПОБУДОВАНОЇ МОДЕЛІ

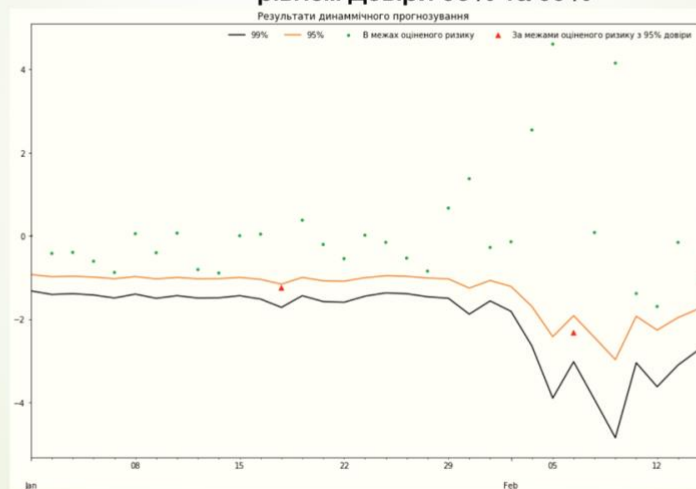
Порівняння значень критеріїв автоматично побудованих моделей ARCH, GARCH, EGARCH

Назва критерію	Модель ARCH	Модель GARCH	Модель EGARCH
MSE	0.168	0.172	0.159
Інформаційний критерій Акаїке	261.4	271.45	252.7
Байєсівський інформаційний критерій	293.8	321.3	283.4
Коефіцієнт детермінації R^2	0.97	0.965	0.974

13

РЕЗУЛЬТАТИ МОДЕЛЮВАННЯ

Результати динамічного прогнозування моделлю EGARCH із урахування оцінки із рівнем довіри 95% та 99%



14

ВИСНОВКИ

В роботі проведено огляд сучасних підходів до моделювання нестационарних процесів та процесів зі змінною в часі дисперсією.

Розроблено програмне забезпечення для побудови авторегресійних моделей, таких як ARIMA, ARCH, GARCH, EGARCH.

В результаті використання прогнозів волатильності реалізовано динамічну оцінку ризиків, результати якої підтверджують здатність моделей до ефективного прогнозування при нестабільній ситуації на ринку.

15



ДЯКУЮ ЗА УВАГУ!